

## DESENVOLVIMENTO DE UM SISTEMA PARA GERENCIAR CURRÍCULOS<sup>1</sup>

**Fernando Mota**

Graduando do 8º período do curso de Sistemas de Informação do UNIPAM.

E-mail: fernandonaylrx@gmail.com

**Fabrcio Araújo**

Professor no curso de Sistemas de Informação do UNIPAM.

E-mail: fabricioa@unipmam.edu.br.

---

**RESUMO:** O presente artigo apresenta o desenvolvimento de um sistema cuja principal funcionalidade é auxiliar na criação de currículos para uma empresa, tendo como propósito ajudar o cliente final na busca e contratação de novos funcionários para sua empresa. Para facilitar o contato entre empresa e candidato, o sistema disponibiliza um *chatbot* no Facebook Messenger para os candidatos criarem seus currículos com o auxílio de um robô. Buscando oferecer uma nova alternativa na busca de novos funcionários, reduzindo custos e criando um ambiente de fácil acesso e utilização para os funcionários das empresas e candidatos a vaga de emprego.

**PALAVRAS-CHAVE:** *Chatbots*; Currículo; Serviços.

**ABSTRACT:** This article presents the development of a system, whose main functionality is help in creation of curriculum for a company, with the purpose of help the end customer find and hire new employees for their company. To facilitate contact between company and candidate, the system provides a chatbot on Facebook Messenger for candidates to create their curriculums with the aid of a robot. Seeking to offer a new alternative in the search of new employees, reducing costs and creating an environment of easy access and use for the employees of the companies and candidates for job vacancy.

**KEYWORDS:** *Chatbots*; Curriculum; Services.

---

### 1 INTRODUÇÃO

As comunicações por meios digitais trouxeram possibilidades que até então não existiam. O surgimento da internet proporcionou que as pessoas pudessem transmitir informações de forma mais rápida e mais interativa. Esta era da comunicação reconfigurou praticamente todos os processos empresariais e muitos processos cotidianos das pessoas, assim como a estrutura da sociedade.

O meio mais comum de se transmitir informações entre pessoas são as redes

---

<sup>1</sup> Trabalho apresentado na área temática 1 - Novas tecnologias e ferramentas para gestão empreendedora do XIV Congresso Mineiro de Empreendedorismo, realizado de 5 a 7 de novembro de 2018.

sociais. Várias pessoas ao redor do mundo as utilizam diariamente, transformando-as em poderosas ferramentas de comunicação, informação, relacionamentos e até mesmo marketing digital. O Facebook, com seus mais de 2 bilhões de usuários, sendo 130 milhões no Brasil, é a rede social mais popular do mundo e do Brasil (COSTA, 2016).

O grande número de usuários e as diversas possibilidades de uso do Facebook são grandes atrativos para empresas. Em 2007, o Facebook disponibilizou APIs (*Application Programming Interface*) para que aplicativos de terceiros pudessem interagir com os usuários da plataforma (KIRKPATRICK, 2007). Desde então, qualquer pessoa com conhecimento em programação consegue criar aplicativos que podem ser usados por qualquer um dos 2 bilhões de usuários do Facebook.

Como já mencionado, perante a era da comunicação vários processos empresariais foram remodelados para alcançar melhores resultados utilizando novas tecnologias. O objetivo deste projeto é melhorar o processo de busca de novos funcionários para uma empresa utilizando o Facebook.

O LinkedIn e distribuidores de ERPs (*Enterprise Resource Planning*) já oferecem serviços para melhorar o processo de busca de novos funcionários. O modo que oferecem seus serviços pode não agradar e atender algumas empresas. As taxas cobradas por esses serviços podem ser um problema para as empresas. O LinkedIn, por exemplo, chega a cobrar até 899 dólares por mês nas suas ferramentas de recrutamento (HUDSON, 2017). Alguns *softwares* ERP possuem designs pouco intuitivos, dificultando trabalhar com eles.

Devido às altas taxas cobradas e os problemas operacionais dos serviços de anúncio de vagas, busca e gestão de novos funcionários, este projeto se justifica através do desenvolvimento de um sistema que atende desde grandes empresas até pequenas e iniciantes. O principal benefício é oferecer uma nova alternativa na busca de novos funcionários que irá reduzir os custos, dar mais visibilidade para os anúncios de vagas e criar um ambiente de fácil acesso e utilização para os funcionários da empresa e os candidatos a vaga.

Para melhorar esse processo, o principal objetivo é desenvolver um sistema baseado em inteligência artificial e aplicações *web*. Será implantado um *chatbot* no Facebook das empresas, onde os candidatos à vaga de emprego enviam seus dados. Estando familiarizados com o *chat* do Facebook, a entrega de informação será mais fluida, dinâmica e direcionada. Utilizando processamento de linguagem natural, o *chatbot* é capaz de conduzir um diálogo para coletar os dados dos usuários e enviar para um serviço que irá gerar currículos padronizados para a empresa.

Para alcançar esse objetivo foram realizadas pesquisas sobre inteligência artificial e algumas de suas áreas como processamento de linguagem natural e o funcionamento de *chatbots*. Após as pesquisas, foi feito o treinamento e os testes do *chatbot*. Foi desenvolvida uma aplicação *web* privada para as empresas gerenciarem os candidatos cujo currículo for cadastrado no Facebook. A troca de dados entre a aplicação *web* e o Facebook foi feita através de APIs.

## 2 REFERENCIAL TEÓRICO

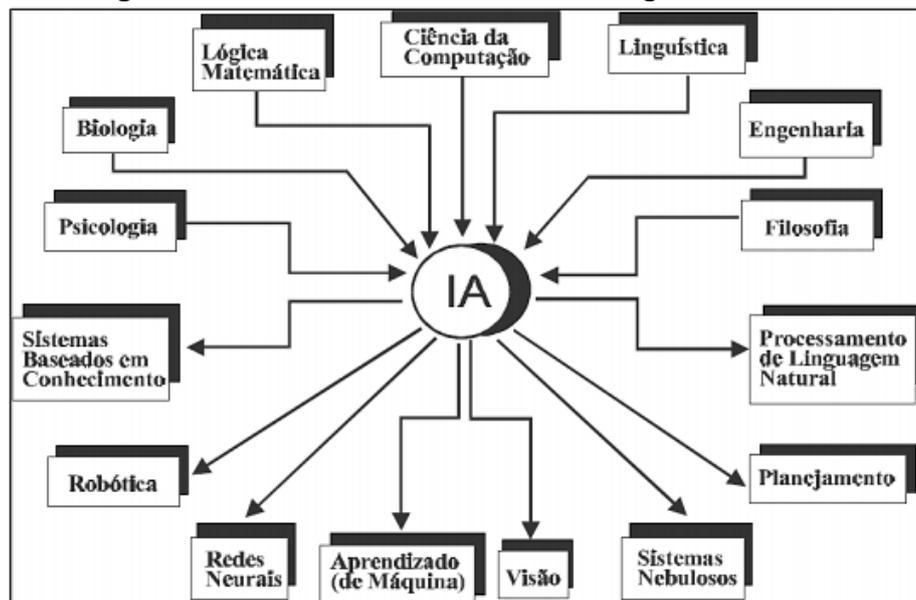
Nesta seção são abordadas técnicas e conceitos que serviram de base para o desenvolvimento do sistema proposto.

### 2.1 INTELIGÊNCIA ARTIFICIAL

IA ou inteligência artificial é um ramo da Ciência da Computação cujo interesse é fazer com que os computadores pensem ou se comportem de forma inteligente, para então conseguirem resolver problemas, jogar jogos, reconhecer padrões, entre outras coisas (JACKSON, 1985).

Com inteligências artificiais diversas resolvendo vários problemas, empresas começaram a usá-la como negócio. Aplicações com IA são inúmeras, elas podem aumentar a eficiência operacional, atuar em gestão de estoques, logística, segurança da informação e otimização de custos (SALOMÃO, 2018). Essa diversidade de aplicações deve-se ao fato de a IA ser um tópico muito extenso como mostra a Figura 1.

**Figura 1 – Áreas Relacionadas com a Inteligência Artificial**



Fonte: GOMES (2011, p. 6).

IA é uma área de pesquisa muito ampla, absorve muito conhecimento de outras áreas como ciência da computação, psicologia e linguística. Por ser uma área muito extensa, acabou gerando outros ramos de pesquisa como *machine learning*, redes neurais e processamento de linguagem natural, a Figura 1 nos descreve isso.

A inteligência artificial é um campo que está sendo pesquisado e aprimorado em grande escala nos últimos anos, além das áreas citadas acima, a IA faz parte de muitas outras áreas, seus conceitos desenvolvidos ao longo dos anos têm trazido grandes benefícios para humanidade.

## 2.2 PROCESSAMENTO DE LINGUAGEM NATURAL

Os seres humanos conseguem entender facilmente frases criadas por outros humanos, por exemplo, “Maria caiu de bicicleta. Ela quebrou uma perna”, sabemos que o “ela” da segunda frase se refere a “Maria”, porque uma bicicleta não tem pernas. Sabemos disso porque aprendemos previamente, mas uma máquina não teve esses ensinamentos, e uma das maneiras de uma máquina aprender é com o processamento de linguagem natural (MORELLATO, 2008).

Processamento de linguagem natural é uma área da IA que estuda a capacidade e as limitações de uma máquina em entender a linguagem dos seres humanos. Seu objetivo é fornecer aos computadores a capacidade de entender textos e comandos de voz (PEREIRA, 2018).

O processamento de linguagem natural consegue entender textos e áudios, pois pega as frases do usuário e caracteriza elas em intenções e filtra suas entidades, para depois entender o que essas frases significam e direcionar a melhor resposta ao usuário. Hoje existem vários serviços que usam esse recurso, os mais famosos e usados são os *chatbots*.

## 2.3 CHATBOT

*Chatbots* são programas especializados em comunicação. Eles podem ser sistemas de conversa simples, interpretando apenas algumas perguntas específicas ou podem ser mais avançado, utilizando recursos de inteligência artificial, no qual é possível desenvolver diálogos em tempo real com a máquina de modo que pareça que esteja conversando com um ser humano (CALADO, 2016).

As vantagens de um *chatbot* são inúmeras, seu desenvolvimento e instalação são muito mais rápidos e barato do que um aplicativo mobile, são facilmente distribuídos, são implantados em interfaces de mensagens de texto que os usuários já estão familiarizados, são extremamente portáteis para diversas plataformas, dentre outras vantagens (GUELBER, 2016).

Com todas as vantagens que os *chatbots* disponibilizam, eles são normalmente utilizados para automatizar o atendimento ao cliente com o intuito de agilizar este processo, podendo ser aplicado também para esclarecer dúvidas e melhorar a maneira de se vender produtos e serviços (ASCARI, 2017). Porém, um *chatbot* não precisa ser restrito somente ao uso dos clientes, eles também podem atender os próprios funcionários, agilizar processos internos, como a entrega de documentos restritos de maneira segura e rápida e resolver tarefas empresariais padronizadas.

No projeto, o *chatbot* foi utilizado como uma ferramenta de entrada dos dados no sistema, utilizando as técnicas de processamento de linguagem natural para receber os dados dos candidatos à vaga de emprego e direcionar respostas ou ações para manter uma conversa fluida e dinâmica.

## 2.4 API

API é uma *interface* simplificada e padronizada, que fornece a vários

programadores um método para construção de aplicações que se comuniquem internamente, fazendo que outros *softwares* consigam utilizar as funcionalidades desta aplicação, sem precisar conhecer detalhes de sua implementação (RIBEIRO, 2016).

Atualmente temos várias APIs públicas que permitem a comunicação entre plataformas. Algumas delas são utilizadas diariamente, como fazer *login* em um aplicativo utilizando sua credencial do Facebook (ANICETO, 2017).

Usar APIs do Facebook possibilitou o *chatbot* comunicar-se com as pessoas pelo Facebook Messenger, facilitando o meio de comunicação entre empresas e candidatos devido à familiarização das pessoas com o Facebook e a portabilidade que o Messenger possui.

### 3 METODOLOGIA

Para adquirir o conhecimento necessário para o desenvolvimento do projeto, foram feitas pesquisas bibliográficas sobre inteligência artificial, criação e utilização de *chatbots*, desenvolvimento e integração de APIs. Para isso foram utilizados livros, artigos e sites referentes aos temas como fonte de pesquisa.

A metodologia utilizada no desenvolvimento do sistema seguiu as práticas do Kanban. A metodologia do Kanban ajudou a assimilar e controlar o fluxo do desenvolvimento de forma prática e contínua utilizando cartões, que representam as diferentes etapas do desenvolvimento do sistema. Esses cartões normalmente possuem três estados, que são a fazer, em andamento e concluído. Ao término de cada etapa, o cartão é puxado para a etapa seguinte até que a mesma seja concluída (BERNARDO, 2014). O gerenciamento do quadro Kanban foi feito através do Trello.

As ferramentas utilizadas para o desenvolvimento deste projeto foram as seguintes:

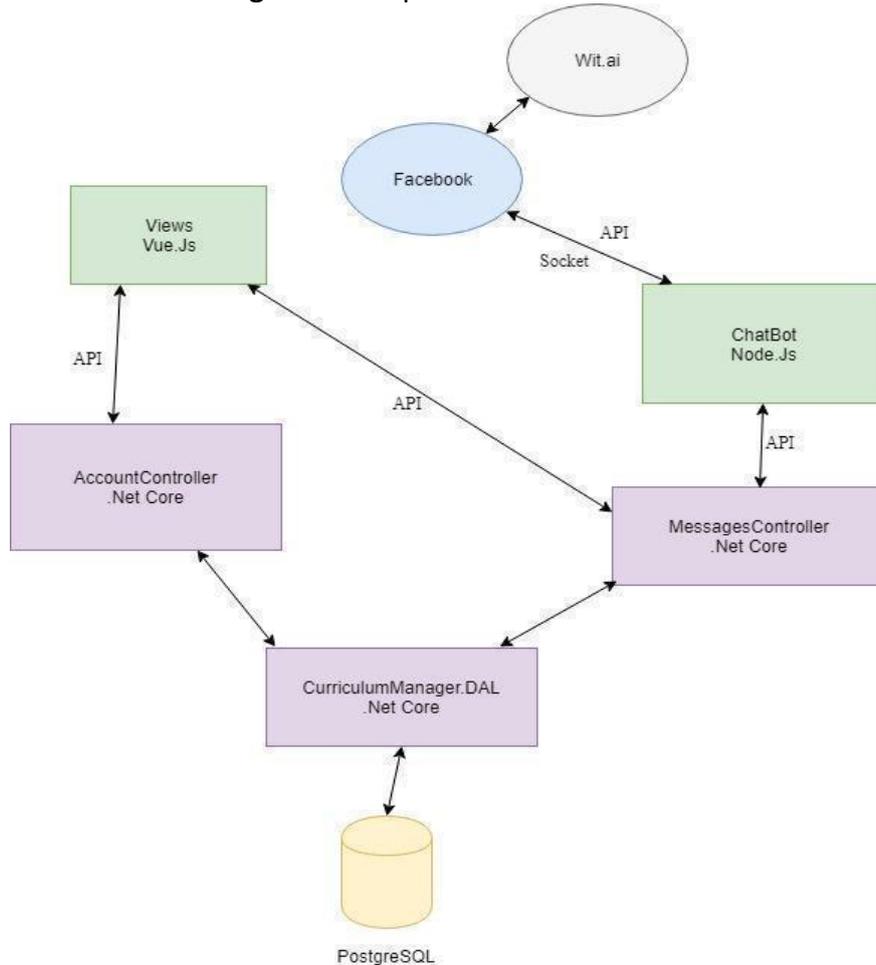
- Vue.JS: *Framework* para a criação de telas para o usuário, utilizado para desenvolver toda a parte do *front-end* da aplicação;
- ASP.Net Core e Node.JS: Estruturas *open source* de alto desempenho para criação de aplicativos, sendo utilizados para criar todo o *back-end* do sistema;
- PostgreSQL: Servidor de banco de dados relacional *open source* utilizado para manipular dados com confiabilidade, robustez e alto desempenho;
- Wit.ai: serviço que utiliza processamento de linguagem natural para identificar as intenções e entidades nas frases enviadas a ele;
- Visual Studio Code: Editor de código-fonte usado para construção de algoritmos;
- Trello: Quadro Kanban gratuito e online, possui várias funcionalidades para gerenciar as tarefas do desenvolvimento do sistema.

O conhecimento adquirido sobre processamento de linguagem natural foi utilizado para o treinamento do *chatbot*. A integração do *chatbot* com o Facebook aconteceu através de APIs que foram sendo desenvolvidas em Node.Js. As APIs desenvolvidas em ASP.Net Core ficaram encarregadas de gerar o currículo para a aplicação *web*. As empresas poderão ver os currículos gerados na parte *web* do projeto através de interfaces gráficas criadas com Vue.Js.

#### 4 DESENVOLVIMENTO E RESULTADOS

Antes de iniciar o desenvolvimento do *software* foi montada uma arquitetura, mostrada na Figura 2. Essa arquitetura seguiu os conceitos do SOA (Arquitetura orientada a serviços). SOA é um padrão de arquitetura com foco na construção e disponibilização de serviços, facilitando a integração e manutenção de sistemas e evitando a replicação dos dados (MENDES, 2013).

**Figura 2 – Arquitetura do Sistema**



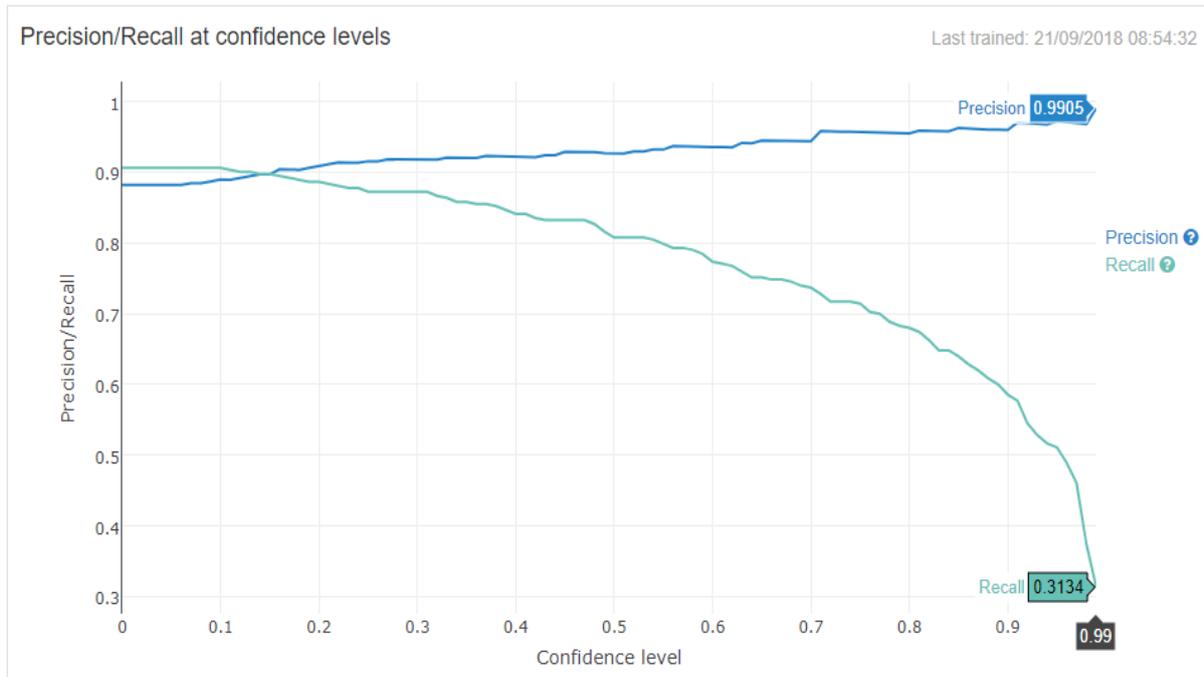
Fonte: Elaborada pelo autor

Cada serviço possui um conjunto de funcionalidades específicas, que de maneira geral fornece recursos a um serviço ou consome recursos de outro serviço. Na Figura 2, cada retângulo é um serviço que foi desenvolvido ao longo do trabalho e as setas indicam a troca de recursos entre eles. Os círculos representam os serviços externos que a aplicação consome.

Nos serviços externos existe o Wit.Ai, que é basicamente uma API gratuita de processamento de linguagem natural que recebe frases e que identifica e retorna as intenções e entidades da frase recebida por meio do seu treinamento. Apesar de não ter a possibilidade de criar uma árvore de diálogo complexa, o Wit consegue conduzir um diálogo simples de perguntas e respostas, que é satisfatório para o projeto.

Foram treinadas ao todo quinze intenções no Wit, todas relacionadas a informações para criar um currículo. Toda vez que é feita uma requisição para o Wit, ele salva qual foi a frase e qual a intenção identificada, para posteriormente ser validada pelo treinador. O Wit disponibiliza um gráfico, que é montado com os dados do *feedback* do treinador, como mostra a Figura 3.

**Figura 3 – Wit Insights**



Fonte: Dados do Wit.AI

O gráfico da Figura 3 mostra duas curvas para uma faixa de níveis de confiança. A curva “*Precision*” mostra com que frequência o Wit está correto quando identifica uma intenção na frase, e a curva “*Recall*” mostra com que frequência o Wit identifica a intenção em uma frase. A curva “*Recall*” apresenta um baixo valor, pois, ao fazer os testes do *chatbot* eram passadas, na maioria das vezes, frases ainda não treinadas. Porém, as frases que já foram treinadas apresentam uma precisão quase perfeita, como mostrado no gráfico da Figura 3.

O Facebook é o outro serviço externo do sistema que envia as mensagens dos candidatos a vaga de emprego para o Wit e ao receber a resposta do Wit com as intenções identificadas o Facebook envia as mensagem e as intenções para o serviço Chatbot, desenvolvido em Node.js, serviço que recebe estas mensagens com as intenções já identificadas. De acordo com a intenção, o serviço direciona a melhor resposta para o candidato, dando continuidade ao diálogo.

Visando um diálogo dinâmico e rápido, as respostas são entregues aos candidatos na forma de texto ou botões. Os botões são usados para responder perguntas onde as respostas serão as mesmas para todos os candidatos, como os exemplos mostrados na Figura 4.

**Figura 4 – Botões do Messenger**

Você é casado ou solteiro?

Casado  
Solteiro

Possui habilitação?

Nenhuma  
CNH-A  
CNH-B

Qual o nível do seu Inglês?

Básico  
Médio  
Avançado

**Fonte:** Elaborada pelo autor

Além de conduzir o diálogo, o serviço Chatbot também é encarregado de buscar os dados do candidato, como, por exemplo, nome e foto, nas APIs do Facebook e repassar esses dados juntamente com as suas mensagens para o serviço que gera o currículo. Os dados em foco são enviados por API.

Os serviços desenvolvidos em .Net Core possuem APIs que utilizam o padrão REST. Os serviços .Net também são divididos em três projetos interligados, sendo eles, API, BLL(*Business Logic Layer*) e DAL(*Data Access Layer*). No projeto API, são localizados os *Controllers* que são responsáveis por responder às solicitações feitas ao serviço por meio de API. No projeto BLL, ficam as regras de negócio da aplicação, como, por exemplo, gerar o currículo, validar se um usuário já está cadastrado no sistema e validar a senha de um usuário. O projeto DAL, que é responsável por fazer todas as operações que envolvem o banco de dados, é compartilhado entre os dois serviços desenvolvidos em .NET.

O sistema de banco de dados utilizado foi o PostgreSQL. Para gerenciar o banco de dados foi utilizada a ferramenta de ORM(*Object Relational Management*) *Entity Framework* e o conceito *Code First*, onde as classes de modelo são escritas primeiro nos serviços em .Net e, em seguida, a partir dessas classes o banco de dados é gerado incluindo os seus relacionamentos, como mostra a Figura 5. O Banco é mantido atualizado conforme as classes são atualizadas, isso se deve à geração e aplicação das *migrations*.

**Figura 5 – Modelando banco de dados em .Net Core**

```
13     protected override void OnModelCreating(ModelBuilder modelBuilder)
14     {
15         modelBuilder.HasDefaultSchema("curriculosDB");
16         modelBuilder.Entity<User>().HasKey(u => u.UserId);
17         modelBuilder.Entity<UserMessage>().HasKey(u => u.IdMessage);
18         modelBuilder.Entity<Sender>().HasKey(u => u.SenderId);
19
20         modelBuilder.Entity<Sender>()
21             .HasMany(c => c.UserMessages)
22             .WithOne(s => s.Sender)
23             .HasForeignKey(b => b.SenderId)
24             .onDelete(DeleteBehavior.Cascade);
25     }
26
```

**Fonte:** Elaborada pelo Autor

O serviço desenvolvido em .Net Core, que recebe os dados do candidato e as mensagens, é o MessagesController. Esse serviço, ao receber um novo candidato, salva os dados do candidato e suas mensagens no banco. Quando é requisitado por API, o serviço pode retornar todo o histórico de conversas do candidato com o *bot*. É nesse serviço que o *front-end* da aplicação *web* para empresas solicita o currículo do candidato.

Para gerar o currículo foi utilizada a biblioteca *ItextSharp*. Essa biblioteca é capaz de criar e escrever arquivos em PDF, podendo editar o tamanho da folha, margens e etc. O PDF é criado com alguns textos estáticos como títulos e subtítulos e também é preenchido com as mensagens do candidato armazenadas no banco. As mensagens são filtradas pelas intenções, ou seja, as mensagens referentes às intenções que forem importantes para o currículo são nele escritas. Ao finalizar o currículo, ele é retornado pela API como um objeto do tipo *MemoryStream*. A Figura 6 mostra um exemplo de currículo criado pelo sistema.

**Figura 6** – Resultado da geração do currículo utilizando *ITextSharp*  
**FERNANDO OLIVEIRA**



Solteiro  
*fernandomota@unipam.edu.br*  
Rua P4A, número 72, bairro Abner Afonso  
+5534991776392

#### **Perfil Profissional**

Trabalho bem em equipe, sou proativo e me adapto fácil a novas demandas e inovações do mercado, além da bagagem teórica de cursos complementares

#### **Formação Acadêmica**

Cursando sistemas de informação no Unipam

#### **Historico Profissional**

Empresa: Unipam

Período: 6 meses

Cargo: Estagiario de desenvolvimento web

Atividades: desenvolvimento de sites, manutenção de computadores, suporte aos usuários

Empresa: Cedro

Período: um ano e quatro meses

Cargo: desenvolvedor mobile

Atividades: desenvolvimento de aplicativo, gerenciamento de servidores e suporte ao cliente

#### **Informações Adicionais**

Habilitação: CNH-B

Nível Inglês: Médio

**Fonte:** Elaborada pelo autor

O serviço AccountController desenvolvido em .Net, tem como função principal gerenciar as contas de usuário da empresa. Nele são feitas requisições via API pelo serviço de *front-end* para criar, editar e excluir usuários. Para garantir uma maior segurança aos dados dos usuários, suas senhas antes de serem salvas no banco são criptografadas com o método *BCrypt*. Também no AccountController que o *front-end* faz requisições por API para fazer *login* no sistema, junto do nome do usuário e senha, também é passado na requisição um *token* que somente os dois serviços conhecem, garantindo mais validação na autenticação do usuário. Ao fazer *login* também é verificado se a senha enviada corresponde à senha criptografada salva no banco.

Os usuários acessam o sistema pelo serviço de *front-end*, o Views, desenvolvido em Vue.Js. Ele funciona com um padrão de componentes, em que cada componente possui suas *tags*, seus *scripts* e seus estilos, transformando cada componente em uma *interface* reaproveitável. Foi utilizado o *framework* Vuetify para criar os componentes

das telas, como menu, formulários, botões e *cards*. O serviço conta com uma tela de *login*, telas para criar e editar contas e uma tela para gerenciar os candidatos, a Figura 7 demonstra exemplos das telas.

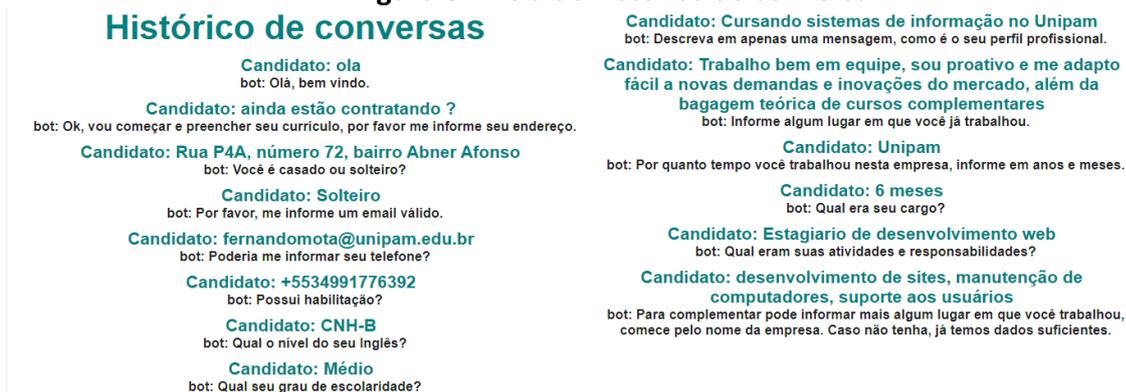
**Figura 7** – Tela de gerenciamento de candidatos



Fonte: Elaborada pelo autor

Ao abrir o site, o cliente é direcionado para a tela de *login*. Caso não possua um usuário, através de um botão no menu, o usuário pode navegar até a página de criar novos usuários. Ao fazer *login*, o cliente é direcionado para a tela de gerenciamento de candidatos. Nessa tela são apresentados todos os candidatos cadastrados dentro de *cards* separados, como mostra a Figura 7. Nesses *cards* são exibidos a foto e o nome do candidato. O *card* também possui três botões, um para ver o histórico de conversas com o *bot*, um exemplo de histórico é mostrado na Figura 8, o outro botão para baixar o PDF com o currículo e, por último, um botão para excluir o candidato do sistema. No menu da tela de gerenciamento de candidatos, há um botão que direciona o usuário para a tela de edição da conta. Nesta tela o usuário pode atualizar seus dados como senha, nome de usuário ou até excluir a conta.

**Figura 8** – Tela de histórico de conversa



Fonte: Elaborada pelo autor

Todas as telas fazem requisições às APIs do *back-end* utilizando o *Axios*. Ele é uma biblioteca para criar um cliente HTTP, podendo assim fazer requisições para as APIs dos outros serviços de maneira muito simples e fácil.

Conforme apresentado nesta seção, o desenvolvimento do sistema foi todo

seguindo o planejamento elaborado a partir da arquitetura criada bem no início do projeto. Esse planejamento garantiu a implementação das funcionalidades de todos os serviços e a integração entre eles.

## 5 CONCLUSÃO

Percebe-se que muitas pessoas tem dificuldade em elaborar um currículo que demonstre todas suas qualidades, essa dificuldade possivelmente pode levar a perda de empregos por parte do candidato. O candidato com mais qualificações e experiências, pode acabar perdendo a vaga para outro candidato que apresente um currículo melhor, mesmo não possuindo as melhores qualificações.

Essa desigualdade de informações nos currículos acaba gerando problemas também no lado das empresas, pois podem estar deixando de fazer a melhor contratação possível. Utilizando o sistema desenvolvido, o processo de seleção se torna mais juntos, pois o currículo seguirá um padrão para todos os candidatos, e os analistas conseguirão analisar currículos completos, com todos os dados necessários para fazer a melhor contratação possível.

Com o desenvolvimento deste trabalho, percebe-se que planejamento antecipado na construção de um *software* só traz benefícios positivos. Possivelmente o sistema teria um melhor desempenho se utilizasse uma base de dados não relacional, como, por exemplo, o MongoDB, pois a maioria dos dados salvos no banco, são dados analíticos e também variam muito na quantidade de memória ocupada. Ao trabalhar com uma arquitetura organizada e definida, uma alta coesão e um baixo acoplamento dos serviços desenvolvidos foram garantidos, facilitando a manutenção do serviço e futuramente o desenvolvimento de novas funcionalidades.

Como melhorias futuras, pode-se citar algumas ideias, como por exemplo o constante aprimoramento do *chatbot*. Sendo possível alcançar um nível em que o *chatbot* seja capaz de conduzir todo o processo de contratação de um funcionário, desde geração de currículo e conduzir uma entrevista com os candidatos até cuidar da parte administrativa da contratação. Adicionar um algoritmo que se baseia em *machine learning* e *deep learning* para prever qual dos novos candidatos seja o melhor para uma vaga de acordo com um banco de dados de currículos de pessoas já contratadas.

Atualmente os processos empresariais estão sempre em evolução e utilizar recursos baseados em inteligência artificial se torna cada vez mais crucial para acompanhar o ritmo da concorrência no mercado. Nos próximos anos, empresas que não se atualizarem e não começarem a utilizar mão de obra automatizada na maioria das tarefas padronizadas e repetitivas, perderão eficiência e, conseqüentemente, clientes, que são a principal fonte de renda de uma empresa.

## REFERÊNCIAS

ANICETO, Caio. **Saiba o que é e para que serve uma API**. 2017. Disponível em: <<https://usemobile.com.br/o-que-e-uma-api/>>. Acesso em 08/abr/2018

ASCARI, Cristina. **Para que serve um chatbot ou bot?** 2017. Disponível em:  
<<https://pt.linkedin.com/pulse/para-que-serve-um-chatbot-ou-bot-cristina-ascari>>.  
Acesso em 03/abr/2018

BAIA, Carlos. **Introdução ao *Machine Learning***. 2016. Disponível em:  
<<http://carlosbaia.com/2016/07/17/introducao-ao-machine-learning/>>. Acesso em  
24/mar/2018

BERNARDO, Kleber. **Kanban: Do início ao fim!** 2014. Disponível em:  
<<https://www.culturaagil.com.br/kanban-do-inicio-ao-fim/>>. Acesso em: 27/mar/2018

CALADO, Caio. **Afinal, o que é chatbot?** 2016. Disponível em:  
<<https://chatbotsbrasil.take.net/afinal-o-que-e-chatbot/>>. Acesso em: 27/mar/2018

COSTA, Thaís. **Quais são as redes sociais mais usadas no Brasil?** 2016. Disponível em:  
<<https://marketingdeconteudo.com/redes-sociais-mais-usadas-no-brasil/>>. Acesso  
em: 11/mar/2018

GOMES, Dennis. **Inteligência Artificial: Conceitos e Aplicações**. 2011. Disponível em:  
<<http://www.olharcientifico.kinghost.net/index.php/olhar/article/viewFile/49/37>>.  
Acesso em: 14/mar/2018

GUELBBER, Filipe. **Criando um robô conversacional inteligente com o Wit.ai**. 2016.  
Disponível em: <<https://labs.bawi.io/criando-um-rob%C3%B4-conversacional-inteligente-com-o-wit-ai-265418fbfd42>>. Acesso em: 27/mar/2018

JACKSON, Philip. **Introduction to artificial intelligence**. 1985. Disponível em:  
<[https://books.google.com.br/books?id=ubviWXXcrvoC&printsec=frontcover&hl=pt-BR&source=gbs\\_ge\\_summary\\_r&cad=0#v=onepage&q&f=false](https://books.google.com.br/books?id=ubviWXXcrvoC&printsec=frontcover&hl=pt-BR&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false)>. Acesso em:  
22/mar/2018

KIRKPATRICK, David. **Facebook's plan to hook up the world**. 2007. Disponível em:  
<<http://archive.fortune.com/2007/05/24/technology/facebook.fortune/index.htm>>.  
Acesso em: 11/mar/2018

MENDES, Edson. **Vantagens e Desvantagens de SOA**. 2013. Disponível em:  
<<https://www.devmedia.com.br/vantagens-e-desvantagens-de-soa/27437>>. Acesso  
em: 24/set/2018

MORELLATO, Luana. **Processamento de Linguagem Natural**. 2008. Disponível em:  
<<https://imasters.com.br/artigo/8537/tendencias/processamento-de-linguagem-natural?trace=1519021197&source=single>>. Acesso em: 27/mar/2018

PEREIRA, Silvio. **Processamento de Linguagem Natural**. 2018. Disponível em:  
<<https://www.ime.usp.br/~slago/IA-pln.pdf>>. Acesso em: 14/mar/2018

RIBEIRO, Marcus. **O que é API e como ele aumenta a produtividade nas empresas.** 2016. Disponível em: <<https://pluga.co/blog/api/o-que-e-api>>. Acesso em: 03/abr/2018

SALOMÃO, Karin. **O Futuro da inteligência artificial e as empresas que a adotaram.** 2018. Disponível em: <<https://exame.abril.com.br/negocios/o-futuro-da-inteligencia-artificial-e-as-empresas-que-a-adotaram/>>. Acesso em: 24/mar/2018