

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

PROTÓTIPO PARA LANÇAMENTO DE NOTAS
ACADÊMICAS UTILIZANDO RECONHECIMENTO DE VOZ

ADRIAN REGUEIRA POSSAMAI

BLUMENAU
2016

ADRIAN REGUEIRA POSSAMAI

**PROTÓTIPO PARA LANÇAMENTO DE NOTAS
ACADÊMICAS UTILIZANDO RECONHECIMENTO DE VOZ**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Sistemas de Informação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Prof. Roberto Heinzle, Doutor - Orientador

**BLUMENAU
2016**

**PROTÓTIPO PARA LANÇAMENTO DE NOTAS
ACADÊMICAS UTILIZANDO RECONHECIMENTO DE VOZ**

Por

ADRIAN REGUEIRA POSSAMAI

Trabalho de Conclusão de Curso aprovado
para obtenção dos créditos na disciplina de
Trabalho de Conclusão de Curso II pela banca
examinadora formada por:

Presidente: _____
Prof. Roberto Heinzle, Doutor – Orientador, FURB

Membro: _____
Prof. Daniel Theisges dos Santos, Mestre – FURB

Membro: _____
Prof. Alexander Roberto Valdameri, Mestre – FURB

Blumenau, 05 de dezembro de 2016

Dedico este trabalho a minha família e a todas as pessoas que me apoiaram e incentivaram nesta caminhada.

AGRADECIMENTOS

A Deus, por me dar forças para concluir esta etapa!

A toda minha família, que me apoiou e incentivou nesta caminhada.

Aos meus amigos, que sempre me apoiaram com palavras de incentivo.

Ao meu orientador Roberto Heinzle, por todo o auxílio prestado.

Aos professores do curso de Sistemas de Informação, pelo conhecimento repassado.

Ao Sr. Joe Elias Linder, por permitir a aplicação deste trabalho.

A todas as pessoas da Edusoft que colaboraram de alguma forma neste trabalho.

O insucesso é apenas uma oportunidade para
recomeçar com mais inteligência.

Henry Ford

RESUMO

Nos últimos anos, com o crescimento da Internet e dos dispositivos móveis, o reconhecimento de voz tem despertado interesse, tornando-se cada vez mais presente em aplicações modernas. Este trabalho apresenta o desenvolvimento de um protótipo para a realização do lançamento de notas acadêmicas utilizando comandos de voz. O protótipo foi desenvolvido utilizando a biblioteca Web Speech API, onde a voz é captada através de um microfone, interpretada pela biblioteca, que retorna em forma de texto. Esse texto é interpretado pelo protótipo que executará as ações solicitadas. A partir dos experimentos realizados, comprovou-se que o protótipo desenvolvido tem a capacidade de reconhecer os comandos mencionados pelo usuário, podendo ser acessado através de um computador ou dispositivo móvel, atingindo os objetivos esperados.

Palavras-chave: Reconhecimento de voz. Web Speech API. Gestão acadêmica.

ABSTRACT

In recent years, with the growth of the Internet and mobile devices, voice recognition has attracted interest, becoming increasingly present in modern applications. This work presents the development of a prototype for a launch of the release of academic notes using voice commands. The prototype was developed using the Web Speech API library, where the voice is captured through a microphone, interpreted by the library, which returns in text form. This text is interpreted by the prototype that will execute the requested actions. From the realized experiments, it was verified that the developed prototype has the capacity to recognize the commands mentioned by the user, being able to be accessed through a computer or mobile device, reaching the expected objectives.

Key-words: Voice recognition. Web Speech API. Academic management.

LISTA DE FIGURAS

Figura 1 - Matriz Distância de Levenshtein	18
Figura 2 - Passo a passo da distância de Levenshtein.....	18
Figura 3 – Tela de <i>login</i> do MentorWeb.....	20
Figura 4– Menu de provas e trabalhos.....	20
Figura 5 – Pesquisa de turmas e disciplinas com ícone para digitação de notas.....	21
Figura 6 – Tela para lançamento das notas das avaliações.....	21
Figura 7 – Avaliação selecionada para o lançamento das notas.....	22
Figura 8 – Notas digitadas e prontas para gravar	22
Figura 9 – Telas do sistema acessadas a partir de um dispositivo móvel	23
Figura 10 – Aguardando comando de voz do usuário	24
Figura 11 – Confirmação do movimento do usuário	25
Figura 12 - Diagrama de casos de uso	27
Figura 13 - Diagrama de classes.....	28
Figura 14 - Menu do MentorWeb.....	35
Figura 15 - Tela de seleção de turma/disciplina.....	36
Figura 16 - Tela para lançamento/consulta de notas.....	36
Figura 17 - Seleção de turma/disciplina.....	48
Figura 18 - Lançamento de notas.....	49

LISTA DE QUADROS

Quadro 1 - Legenda da figura “Passo a passo da distância de levenshtein”	18
Quadro 2 - Comparativo entre os trabalhos correlatos	25
Quadro 3 - Principais requisitos funcionais do protótipo	26
Quadro 4 - Principais requisitos não funcionais do protótipo	26
Quadro 5 - Inicializando o reconhecimento de fala	30
Quadro 6 - Identificando o resultado final obtido	31
Quadro 7 - Palavras chaves aplicáveis	31
Quadro 8 - Identificando a palavra chave ou valor que a sucede (método: realizaReconhecimento)	32
Quadro 9 - Processando palavra chave, com pausa (método: realizaReconhecimento)	33
Quadro 10 - Processando palavra chave, sem pausa (método: realizaReconhecimento)	34
Quadro 11 - Algoritmo de Levenshtein em Java	35
Quadro 12 - Experimento 01: validação da busca dos alunos a partir do nome	37
Quadro 13 - Experimento 02: validação do lançamento de notas	38
Quadro 14 - Perfil de usuários que realizaram o teste de usabilidade	39
Quadro 15 - Respostas referentes ao questionário de usabilidade	40
Quadro 16 - UC01 - Selecionar turma/disciplina	44
Quadro 17 – UC02 - Selecionar período de nota	44
Quadro 18 - UC03 - Selecionar avaliação	45
Quadro 19 - UC04 - Selecionar forma de lançamento de notas	45
Quadro 20 - UC05 Realizar o lançamento de notas	46
Quadro 21 - UC06 Consultar notas	46
Quadro 22 - UC07 - Alterar notas registradas por qualquer professor	47
Quadro 23 - Atributos fundamentais das principais classes do prototipo	50
Quadro 24 - Questionário 01 - Perfil do usuário	51
Quadro 25 - Instruções para a avaliação do usuário	52
Quadro 26 - Questionário 02 - Teste de usabilidade	53

LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface

HTML – HyperText Markup Language

IHC – Interface Humano Computador

RF – Requisitos funcionais

RNF – Requisitos não funcionais

SUMÁRIO

1 INTRODUÇÃO	13
1.1 OBJETIVOS.....	13
1.2 ESTRUTURA.....	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 FERRAMENTAS PARA GESTÃO ACADÊMICA.....	15
2.2 RECONHECIMENTO DE VOZ.....	15
2.3 RECONHECIMENTO DE VOZ COM WEB SPEECH API.....	16
2.4 DESIGN RESPONSIVO.....	17
2.5 ALGORITMO DISTÂNCIA DE LEVENSHTein.....	17
2.6 SISTEMA ATUAL.....	19
2.6.1 Módulo Secretaria.....	19
2.6.2 Módulo Central do Professor	19
2.7 TRABALHOS CORRELATOS	23
2.7.1 Voz da Notícia: Aplicativo web de notícias para deficientes visuais.....	23
2.7.2 Desenvolvimento de uma plataforma para um jogo de xadrez inteligente controlado por reconhecimento de voz.....	24
2.7.3 Comparativo entre os aspectos dos trabalhos correlatos.....	25
3 DESENVOLVIMENTO DO PROTÓTIPO	26
3.1 PRINCIPAIS REQUISITOS DO PROTÓTIPO	26
3.2 ESPECIFICAÇÃO.....	26
3.2.1 Diagrama de casos de uso	27
3.2.2 Diagrama de classe	27
3.3 IMPLEMENTAÇÃO	29
3.3.1 Técnicas e ferramentas utilizadas	29
3.3.2 Etapas para realizar o reconhecimento de voz	30
3.3.3 Implementação do Algoritmo de Levenshtein	34
3.3.4 Operacionalidade do protótipo	35
3.4 RESULTADOS E DISCUSSÕES	37
3.4.1 Experimento 01: validar a busca dos alunos a partir do nome.....	37
3.4.2 Experimento 02: validar o lançamento de notas.....	38
3.4.3 Experimento 03: validar o lançamento de notas em um dispositivo móvel.....	39

3.4.4 Experimento 04: teste de usabilidade	39
4 CONCLUSÕES	41
4.1 EXTENSÕES	41
APÊNDICE A – DESCRIÇÃO DOS CASOS DE USO.....	44
APÊNDICE B – TELAS DO PROTÓTIPO EM VERSÃO MOBILE.....	48
APÊNDICE C – ATRIBUTOS FUNDAMENTAIS DAS PRINCIPAIS CLASSES DO PROTÓTIPO.....	50
APÊNDICE D – ROTEIRO E QUESTIONÁRIO DE AVALIAÇÃO DA USABILIDADE.....	51

1 INTRODUÇÃO

Nos últimos anos com o crescimento gigantesco da Internet e dos dispositivos móveis, o reconhecimento de voz têm despertado interesse, tornando-se cada vez mais presente em aplicações modernas (ALENCAR; ALCAIM, 2008). Segundo Ynoguti (1999), as interfaces via comando de voz estão se tornando uma necessidade, pois é a forma mais natural da comunicação humana, tornando-a mais simples e interativa.

Atualmente o crescimento no uso de dispositivos móveis fez com que muitos usuários adviessem a utilizá-los com frequência para acesso à sites e aplicativos diversos. Entretanto, muitos destes aplicativos foram originalmente projetados apenas para acesso via computadores, tornando-os praticamente inacessíveis via dispositivo móvel (KLERING, 2015). Além do crescimento das aplicações que utilizam o reconhecimento de voz, surgiu um novo paradigma para o desenvolvimento de aplicações web, chamado design responsivo. O design responsivo tem como objetivo principal adequar um mesmo leiaute a várias resoluções de tela, garantindo a boa experiência do usuário independente do dispositivo que está sendo utilizado (DAUGS, 2016).

A Edusoft é a empresa responsável pelo desenvolvimento do software de gestão acadêmica MentorWeb. O software desenvolvido pela empresa tem o objetivo de informatizar instituições de ensino, independente do porte. O mesmo oferece recursos para a completa administração de uma escola, incluindo tanto a gestão acadêmico-pedagógica como a gestão administrativo-financeira. Porém, a maior parte do sistema não é homologada para acesso via dispositivo móvel. Segundo Souza (2015), dentre as solicitações que a empresa recebe por parte dos clientes, enfatiza-se que o lançamento de notas, realizado pelo docente. Trata-se de uma das principais funcionalidades que necessitariam ser disponibilizadas para acesso independente do dispositivo.

Com base na afirmação de Souza (2015), este trabalho de conclusão de curso propõe o desenvolvimento de um protótipo que permita realizar o lançamento de notas de uma forma interativa utilizando comando de voz, e adequando o leiaute da tela de acordo com o dispositivo que está realizando o acesso.

1.1 OBJETIVOS

O objetivo geral do trabalho é a construção de um protótipo para o lançamento de notas acadêmicas no MentorWeb utilizando reconhecimento de voz.

Os objetivos específicos do trabalho são:

- a) capturar o áudio com o auxílio de um microfone;

- b) integrar a biblioteca de reconhecimento de voz Web Speech API com o protótipo;
- c) utilizar técnicas e algoritmos de comparação de palavras para identificar os alunos na tela de lançamento de notas do protótipo.

1.2 ESTRUTURA

Este trabalho está organizado na forma de capítulos. O primeiro capítulo apresenta a introdução, a justificativa e os objetivos deste trabalho.

No segundo capítulo tem-se a fundamentação teórica, onde são apresentadas as utilidades de ferramentas para a gestão de instituições de ensino, os conceitos sobre reconhecimento de voz e design responsivo, o sistema atual e a apresentação dos trabalhos correlatos.

O terceiro capítulo destaca o desenvolvimento do protótipo, levantamento de requisitos, conceituações de técnicas e ferramentas utilizadas, a implementação do protótipo e, por fim, os resultados obtidos.

No quarto capítulo, apresentam-se as conclusões, bem como sugestões para futuros trabalhos.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo está organizado em sete seções. A seção 2.1 descreve as ferramentas para a gestão acadêmica. A seção 2.2 apresenta os conceitos sobre reconhecimento de voz, complementada pela seção 2.3, que apresenta o reconhecimento de voz com Web Speech API. Na seção 2.4 conceitua-se o design responsivo e sua utilidade. Já na seção 2.5 aborda-se o algoritmo de Levenshtein, esboçando seu funcionamento. A seção 2.6 apresenta o sistema atual, ilustrando as telas do processo de lançamento de notas. Por fim, a seção 2.7 descreve os trabalhos correlatos.

2.1 FERRAMENTAS PARA GESTÃO ACADÊMICA

As ferramentas de gestão acadêmica melhoram o dia a dia das instituições de ensino, prestando auxílio a gestores, professores e alunos. Muitas dessas ferramentas possuem recursos para gerenciar toda a instituição, tanto a parte acadêmico-pedagógica bem como administrativo-financeira. Os softwares mais modernos permitem ao professor o registro de frequência e lançamento de notas via internet, evitando a necessidade de carregar pilhas de papéis (PORVIR, 2015).

Segundo Moran (2003), uma instituição de ensino, por menor que seja, precisa possuir uma forma bem definida e adaptada à sua realidade para gerenciar seus dados. Os principais colégios e universidades brasileiras já utilizam ferramentas que auxiliam na gestão acadêmica. Essas ferramentas têm como objetivo principal integrar todas as informações referentes à instituição de ensino, diminuindo a quantidade de papéis e arquivos físicos, facilitando assim a recuperação de informações quando necessário.

Até pouco tempo, pais e alunos precisavam ir à instituição de ensino para ter acesso às informações, tais como: boletins, ocorrências, históricos escolares e segunda via de boletos. Hoje, com o advento da tecnologia, essas informações podem ser acessadas através da internet, a qualquer hora e de qualquer lugar (EDUSOFT, 2014a).

2.2 RECONHECIMENTO DE VOZ

Segundo Ynoguti (1999), um sistema de reconhecimento de voz converte o sinal acústico observado em sua representação ortográfica correspondente. O sistema faz a sua escolha a partir de um vocabulário finito de palavras que podem ser reconhecidas. Por simplicidade, assume-se que uma palavra é identificada somente por sua pronúncia.

Os estudos sobre reconhecimento de fala iniciaram-se em 1950, onde os pesquisadores que procuravam estudar a fonética-acústica não obtiveram bons resultados. A partir do ano de 1960 intensificaram-se as pesquisas referentes ao reconhecimento de voz. Em 1980 foram obtidos melhores resultados, com a aplicação de redes neurais ao reconhecimento de fala (OLIVEIRA, 2002).

Atualmente, vários softwares no mercado estão utilizando recursos de reconhecimento de voz. Um exemplo são os sistemas de atendimento eletrônico. Eles interpretam o som emitido pela voz, classificam em sílabas e aplicam um método de busca para associar essas informações à padrões de palavras com o objetivo de encontrar semelhanças. Em smartphones, é comum esse tipo de tecnologia em aplicativos que processam sílabas de músicas e realizam a identificação de nome e cantor (PEREIRA, 2009).

A empresa Google tem investido nesta tecnologia. Introduziu a biblioteca Web Speech API em seu navegador (Google Chrome), tornando possível o reconhecimento de fala (LIMER, 2013). Com isso, para os usuários que utilizam o navegador Google Chrome, está disponível o recurso de pesquisa por voz no próprio buscador da empresa.

2.3 RECONHECIMENTO DE VOZ COM WEB SPEECH API

O Web Speech API tem por objetivo permitir a entrada de textos em formulários HyperText Markup Language (HTML) utilizando a fala, possibilitando realizar essa entrada de forma contínua. A Application Programming Interface (API) processa os sinais de voz, transformando as ondas sonoras em texto. Desta forma, obtêm-se resultados que são enviados para a página web como uma lista de hipóteses, juntamente com outras informações relevantes para cada uma delas (SHIRES; WENNBORG, 2012).

Segundo Oliveira (2014), o Web Speech API é desenvolvido na linguagem Javascript e implementado no navegador Google Chrome na versão 25 ou superior. Essa API fornece uma biblioteca de funções que possibilitam a interação do usuário com a máquina por meio de recursos de fala.

Vieira (2014) destaca que o Web Speech API permite que desenvolvedores web incorporem o reconhecimento de fala em suas aplicações, possibilitando reconhecer a voz do usuário a partir de uma entrada de áudio e transformá-la em texto. Segundo ele, o áudio é enviado para o servidor remoto, que processa e retorna para o navegador em forma de texto. O texto gerado é uma hipótese do que foi reconhecido, pois não há uma garantia de que esteja correto. Shires e Wennborg (2012) descrevem que o Web Speech API retorna junto à lista de

palavras reconhecidas, um atributo que representa o nível de confiança desse reconhecimento. Quanto maior o valor deste atributo, maior é a possibilidade do texto retornado estar correto.

2.4 DESIGN RESPONSIVO

Segundo Daus (2016), o conceito sobre design responsivo surgiu em 2010, com o objetivo de adequar um mesmo leiaute a várias resoluções de tela. Essa tecnologia permite que o usuário tenha alta qualidade acessando a página web, independente do dispositivo utilizado.

Moura (2016) define design responsivo como um sistema que possui a capacidade de ajustar a disposição de seus componentes visuais, adaptando-se à resolução de tela na qual é exibido, alterando a altura, largura, posição ou até mesmo a formatação de elementos gráficos, com o objetivo de garantir melhor usabilidade ao usuário. Segundo Moura (2016), o Google implementou um novo algoritmo em seu site de buscas, que possui a função de avaliar, por meio de critérios matemáticos, a navegabilidade dentro de uma página web. Quanto melhor a sua navegabilidade, melhor é a sua posição no ranking de resultados nas pesquisas realizadas.

Antes do advento do design responsivo, o leiaute de uma página web era desenvolvido com base em pixel. Porém, essa técnica de desenvolvimento não permite que um único leiaute se adapte a diversas resoluções de tela, por ocupar um espaço fixo. Para desenvolver um leiaute responsivo, o conceito de pixel deve ser deixado de lado, devendo-se trabalhar a partir de porcentagens. Por exemplo, ao invés de definir que determinado componente ocupará duzentos pixels da tela, pode-se delimitar o seu tamanho em cinco por cento do tamanho dela. Ou seja, independente da resolução utilizada, o componente ocupará somente a porcentagem definida (GUERRATO, 2014).

2.5 ALGORITMO DISTÂNCIA DE LEVENSHTAIN

Segundo Balsini (2012), o algoritmo Levenshtein Distance (distância de Levenshtein) foi criado em 1966 pelo cientista Vlademir Levenshtein. O algoritmo tem por objetivo verificar a similaridade entre duas cadeias de caracteres. A distância de Levenshtein é obtida através do menor número de interações, substituições ou eliminações de caracteres necessários para transformar uma palavra em outra. Quanto mais próximo de zero for o resultado obtido, mais próximas são as palavras. Se o resultado for igual a zero, significa que as cadeias de caracteres analisadas são exatamente iguais (SILVA, 2007).

A partir da quantidade total de caracteres dos textos comparados é realizada a montagem de uma matriz, onde serão inseridos pesos para cada operação necessária para igualar as palavras. A matriz pode ser preenchida começando no canto superior esquerdo em direção ao canto inferior direito. Cada salto horizontal na matriz significa que houve uma inserção ou substituição de algum caractere. Cada salto vertical indica uma exclusão, e valores iguais na diagonal, significa uma equivalência (BALSINI, 2012).

A Figura 1 ilustra a matriz utilizada pelo algoritmo de Levenshtein, mostrando a simulação da comparação entre as palavras “levenshtein” e “meilenshtein”, onde foram necessários quatro passos para igualá-las. A Figura 2 mostra o passo a passo do algoritmo, e por fim, o Quadro 1 apresenta a legenda referente a esta figura.

Figura 1 - Matriz Distância de Levenshtein

		m	e	i	l	e	n	s	t	e	i	n
	0	1	2	3	4	5	6	7	8	9	10	11
l	1	1	2	3	3	4	5	6	7	8	9	10
e	2	2	1	2	3	3	4	5	6	7	8	9
v	3	3	2	2	3	4	4	5	6	7	8	9
e	4	4	3	3	3	3	4	5	6	6	7	8
n	5	5	4	4	4	4	3	4	5	6	7	7
s	6	6	5	5	5	5	4	3	4	5	6	7
h	7	7	6	6	6	6	5	4	4	5	6	7
t	8	8	7	7	7	7	6	5	4	5	6	7
e	9	9	8	8	8	7	7	6	5	4	5	6
i	10	10	9	8	9	8	8	7	6	5	4	5
n	11	11	10	9	9	9	8	8	7	6	5	4

Fonte: Balsini (2012, p. 32).

Figura 2 - Passo a passo da distância de Levenshtein

l	e	v	e	n	s	h	t	e	i	n	or	l	e	v	e	n	s	h	t	e	i	n
o	=	+	o	=	=	=	=	=	=	=		o	=	o	+	=	=	=	=	=	=	=
m	e	i	l	e	n	s	t	e	i	n		m	e	i	l	e	n	s	t	e	i	n

Fonte: Balsini (2012, p. 32).

Quadro 1 - Legenda da figura “Passo a passo da distância de levenshtein”

Símbolo	Legenda
=	Caractere equivalente
O	Substituição de caractere
+	Inserção de caractere
-	Exclusão de caractere

Fonte: elaborado pelo autor.

2.6 SISTEMA ATUAL

A Edusoft Tecnologia é uma empresa que desenvolve softwares para gestão acadêmica. Hoje, possui cerca de mil unidades de ensino como clientes em treze estados brasileiros, somando mais de trezentos mil alunos ativos registrados em seus sistemas. Segundo Edusoft (2014b), a missão da empresa é desenvolver sistemas de informação para gestão acadêmica com as melhores tecnologias, elevando a competitividade de seus clientes, e permitindo alcançar a liderança em seu nicho de mercado.

Atualmente, o MentorWeb, software de gestão acadêmica desenvolvido pela Edusoft, é totalmente web. O sistema é dividido em vários módulos que auxiliam na gestão da instituição de ensino, desde a área acadêmica até a parte financeira e contábil. O acesso aos módulos do sistema e suas funcionalidades são definidos de acordo com o perfil de usuário. Os perfis de usuário são definidos pela instituição, disponibilizando para cada perfil somente as funcionalidades para os quais têm acesso. Como exemplo de perfis de usuário, pode-se citar administrador do sistema, secretária, professor, coordenador e aluno. Dentre os principais módulos do sistema, há dois que mantêm relação estreita com este trabalho: secretaria e a central do professor (EDUSOFT, 2012).

2.6.1 Módulo Secretaria

O módulo secretaria é um dos mais importantes do MentorWeb. Reúne uma série de cadastros, operações e controles que envolvem todo o processo acadêmico. Dentre as principais funcionalidades deste módulo estão o cadastro de curso, grade curricular, período letivo, turma, disciplina e realização da matrícula de novos alunos (EDUSOFT, 2012).

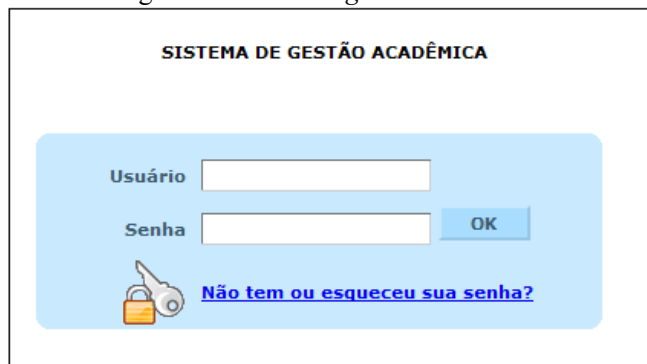
Neste módulo, realiza-se a maioria das configurações referentes aos demais módulos do sistema. Para a central do professor, configura-se, por exemplo, o período de lançamento de notas das avaliações e o cronograma de aula. A partir deste módulo, pode-se também alterar as informações registradas pelos docentes (EDUSOFT, 2012).

2.6.2 Módulo Central do Professor

Outro módulo do MentorWeb muito utilizado é a central do professor. Dentre as principais funcionalidades do módulo, destacam-se: realizar o registro do plano de ensino, cadastro e agendamento de avaliações, registro de conteúdo ministrado, frequência diária, e, uma das mais utilizadas, o lançamento de notas (EDUSOFT, 2012). Hoje, o lançamento das notas das avaliações é realizado da seguinte maneira:

- a) o professor realiza o *login* na central do professor, conforme mostra a Figura 3;

Figura 3 – Tela de *login* do MentorWeb



A tela de login do MentorWeb apresenta o título "SISTEMA DE GESTÃO ACADÊMICA" no topo. Abaixo, há um formulário com dois campos de entrada: "Usuário" e "Senha". Um botão "OK" está à direita do campo "Senha". Abaixo dos campos, há um ícone de uma chave e um cadeado, seguido pelo link "Não tem ou esqueceu sua senha?".

Fonte: elaborado pelo autor.

- b) em seguida, no canto lateral esquerdo da tela, visualizará o menu principal. O professor clica no link *Provas e Trabalhos*, conforme ilustra a Figura 4;

Figura 4– Menu de provas e trabalhos



Fonte: elaborado pelo autor.

- c) neste momento, o sistema apresenta uma tela com filtros para pesquisa de turmas e disciplinas. Após informá-los, o professor clica no botão *F9-Pesquisar* para listar as turmas e disciplinas encontradas, contendo uma coluna com o ícone de *Provas e trabalhos* para realizar o lançamento das notas, conforme mostra a Figura 5;

Figura 5 – Pesquisa de turmas e disciplinas com ícone para digitação de notas

Como pesquisar?

Período letivo: 2014/2

Curso: [dropdown]

Situação da turma: Em andamento

Diário eletrônico de 1 até 1 total: 1

Turma/Disciplina	Curso	Provas e trabalhos
20142/C01F02 / Contabilidade (Fundamentos para Administradores)	Administração	

Fonte: elaborado pelo autor.

- d) após clicar no ícone Provas e trabalhos referente à turma e disciplina escolhida, o sistema apresenta a tela para lançamento das notas, listando todos os alunos matriculados, com as respectivas avaliações aplicadas, conforme ilustra a Figura 6;

Figura 6 – Tela para lançamento das notas das avaliações

F10-Gravar

Período letivo: 2014/2

Curso: 01 - Administração

Turma: 20142/C01F02

Disciplina: 1809 - Contabilidade

Fase: [dropdown]

Fórmula de cálculo: Fórmula por pesos

Avaliação: Nota 1

Visualizar foto:

somente alunos ativos:

N.º	Aluno	AVA02 03/09/2014	AVA01 10/09/2014
2	Alan Silva	8,00	
7	Aluno da instituição		
6	Etomate Lopes		
4	Lilian Maria	8,50	
1	Sandro Maradona	8,40	
3	Taise Souza	8,00	

Fonte: elaborado pelo autor.

- e) o professor marca a avaliação para qual deseja realizar o lançamento das notas, e o sistema habilita as caixas de texto para permitir que as notas sejam informadas, conforme mostra a Figura 7.

Figura 7 – Avaliação selecionada para o lançamento das notas

F10-Gravar			
Período letivo 2014/2	Curso 01 - Administração	Turma 20142/C01F02	
Disciplina 1809 - Contabilidade	Fase	Visualizar foto <input type="checkbox"/>	
Fórmula de cálculo Fórmula por pesos	Avaliação Nota 1	somente alunos ativos <input checked="" type="checkbox"/>	
N.º	Aluno	AVA02 03/09/2014	AVA01 10/09/2014
		<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Alan Silva	<input type="text"/>	<input type="text"/>
7	Aluno da instituição	<input type="text"/>	<input type="text"/>
6	Etomate Lopes	<input type="text"/>	<input type="text"/>
4	Lilian Maria	<input type="text"/>	<input type="text"/>
1	Sandro Maradona	<input type="text"/>	<input type="text"/>
3	Taise Souza	<input type="text"/>	<input type="text"/>

Fonte: elaborado pelo autor.

- f) para cada avaliação corrigida, o docente procura o aluno na tela, clica na caixa de texto respectiva, e realiza a digitação da nota;
- g) após realizar o lançamento das notas, o docente clica no botão F10 - Gravar, para gravar os registros informados, conforme ilustra a Figura 8.

Figura 8 – Notas digitadas e prontas para gravar

F10-Gravar			
Período letivo 2014/2	Curso 01 - Administração	Turma 20142/C01F02	
Disciplina 1809 - Contabilidade	Fase	Visualizar foto <input type="checkbox"/>	
Fórmula de cálculo Fórmula por pesos	Avaliação Nota 1	somente alunos ativos <input checked="" type="checkbox"/>	
N.º	Aluno	AVA02 03/09/2014	AVA01 10/09/2014
		<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	Alan Silva	10,00	<input type="text"/>
7	Aluno da instituição	9,75	<input type="text"/>
6	Etomate Lopes	7,00	<input type="text"/>
4	Lilian Maria	8,95	<input type="text"/>
1	Sandro Maradona	10,00	<input type="text"/>
3	Taise Souza	7,85	<input type="text"/>

Fonte: elaborado pelo autor.

2.7 TRABALHOS CORRELATOS

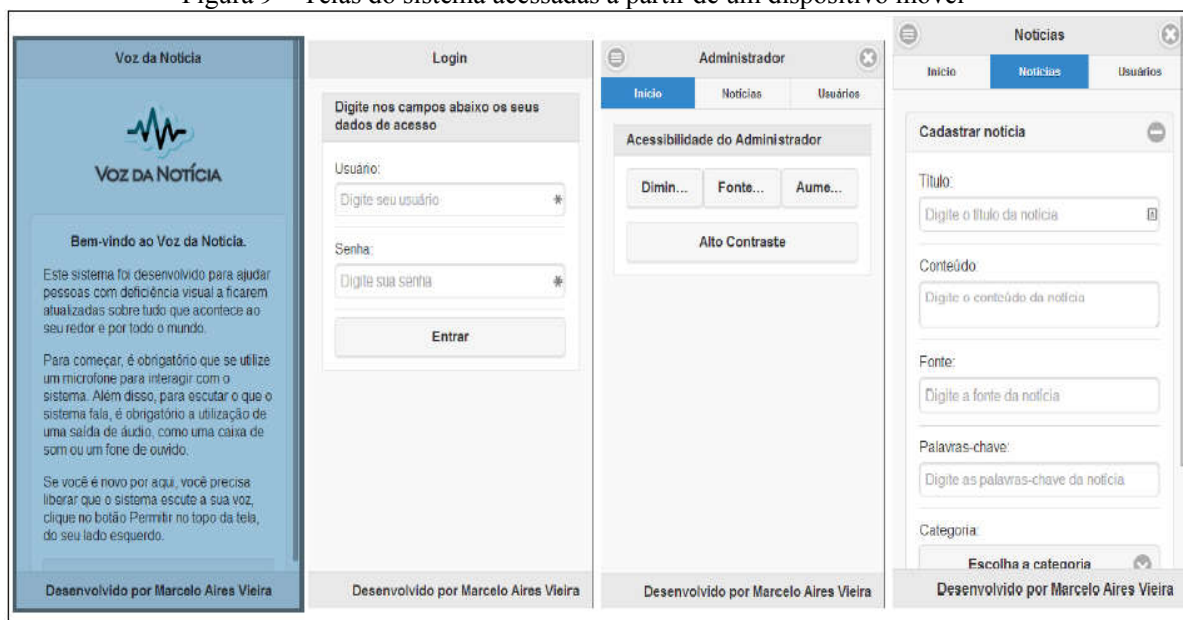
O objetivo desta seção é apresentar alguns trabalhos que utilizaram o Web Speech API para realizar o reconhecimento de voz. Foram encontrados os trabalhos de Vieira (2014) e Oliveira (2014).

2.7.1 Voz da Notícia: Aplicativo web de notícias para deficientes visuais

Vieira (2014) desenvolveu um aplicativo web de notícias, com o objetivo de ser utilizado por deficientes visuais. A ferramenta foi desenvolvida seguindo técnicas de acessibilidade e usabilidade de Interface Humano Computador (IHC). Para o desenvolvimento do aplicativo, utilizou-se a linguagem de programação PHP, integrando a biblioteca Web Speech API para transformar texto em fala e vice-versa.

O aplicativo desenvolvido é voltado principalmente para deficientes visuais. Ele permite que o usuário possa ouvir a notícia ao invés de ler, com a possibilidade de interagir com o sistema, a partir da sua própria voz. Por este motivo, possui poucas interfaces, sendo elas ajustáveis a diversos tamanhos de telas, podendo ser acessadas em dispositivos móveis ou computadores. A Figura 9 ilustra algumas interfaces da ferramenta, acessada a partir de um dispositivo móvel.

Figura 9 – Telas do sistema acessadas a partir de um dispositivo móvel



Fonte: Vieira (2014, p. 48).

Segundo o autor, após a implementação foram realizados vários testes para avaliar a usabilidade e acessibilidade do aplicativo. Com isso, concluiu-se que o deficiente visual

conseguiu utilizar a ferramenta sem dificuldade, permitindo pesquisar notícias novas de forma acessível e interativa.

2.7.2 Desenvolvimento de uma plataforma para um jogo de xadrez inteligente controlado por reconhecimento de voz

Oliveira (2014) propôs uma plataforma visual para realizar partidas de jogos de xadrez contra o computador. O aplicativo controla os movimentos realizados pelo usuário utilizando reconhecimento de voz, e os do computador, utilizando inteligência artificial. Para o desenvolvimento do aplicativo, foi utilizada a linguagem de programação Javascript, integrando a biblioteca Web Speech API para interpretar os comandos de voz.

Para iniciar o jogo, o usuário deve clicar no ícone `microfone`. Neste momento, o aplicativo fica aguardando o usuário informar o movimento a ser realizado no jogo, conforme mostra a Figura 10.

Figura 10 – Aguardando comando de voz do usuário



Fonte: Oliveira (2014, p. 46).

Enquanto o usuário fala, o aplicativo insere na a caixa de texto (ao lado esquerdo do ícone `microfone`) a descrição movimentos mencionados por ele. Ao término, o usuário clica novamente no ícone `microfone` para confirmar seu movimento. Neste momento, o aplicativo valida o movimento do usuário, e confirma sua jogada, emitindo uma mensagem de alerta em tela, mostrando que a jogada foi bem sucedida, conforme ilustra a Figura 11.

Figura 11 – Confirmação do movimento do usuário



Fonte: Oliveira (2014, p. 47).

Com isso, o autor afirma que no uso do aplicativo, o usuário consegue jogar xadrez, realizando suas jogadas por meio de comandos de voz.

2.7.3 Comparativo entre os aspectos dos trabalhos correlatos

O Quadro 2 demonstra o comparativo das principais características dos trabalhos citados na seção 2.7.

Quadro 2 - Comparativo entre os trabalhos correlatos

Aspectos dos trabalhos correlatos	Vieira (2014)	Oliveira (2014)
Utiliza Web Speech API	Sim	Sim
Homologado para dispositivo móvel	Sim	Não
Capacidade de se ajustar a diversos tamanhos de tela	Sim	Não
Linguagem de programação predominante	PHP	Javascript

Fonte: elaborado pelo autor.

Com base no Quadro 2, pode-se concluir que ambos os trabalhos correlatos utilizaram o Web Speech API para realizar o reconhecimento de voz, embora utilizem linguagens de programação diferentes. Porém, o trabalho de Vieira (2014) possui a capacidade de se ajustar a diversos tamanhos de tela, já o de Oliveira (2014), está homologado apenas para computadores.

A relação entre o trabalho de Vieira (2014) e o presente trabalho é o fato da utilização da biblioteca Web Speech API, para possibilitar o reconhecimento de voz, e a capacidade de se ajustar a diversos tamanhos de tela. Já a relação com o trabalho de Oliveira (2014), além da utilização do Web Speech API, também se dá pelo fato da utilização da linguagem Javascript.

3 DESENVOLVIMENTO DO PROTÓTIPO

Nesse capítulo são apresentadas as etapas do desenvolvimento do protótipo. A seção 3.1 apresenta os principais requisitos do trabalho. Na seção 3.2 apresenta-se a especificação, com diagramas de casos de uso e diagrama de classe. A seção 3.3 contém a implementação com as técnicas e ferramentas utilizadas, além da operacionalidade do protótipo. Por fim, a seção 3.4 expõe os resultados obtidos.

3.1 PRINCIPAIS REQUISITOS DO PROTÓTIPO

O Quadro 3 apresenta os principais requisitos funcionais (RF) do protótipo. O Quadro 4 apresenta os principais requisitos não funcionais (RNF) que interferem diretamente no funcionamento do mesmo.

Quadro 3 - Principais requisitos funcionais do protótipo

Requisitos Funcionais (RF)	
RF01	o sistema deverá permitir ao usuário selecionar a turma/disciplina para realizar o lançamento de notas.
RF02	o sistema deverá apresentar os alunos matriculados na turma/disciplina selecionada.
RF03	o sistema deverá permitir ao usuário selecionar o período de notas para o qual serão lançadas as notas.
RF04	o sistema deverá carregar as respectivas avaliações do período de nota selecionado.
RF05	o sistema deverá permitir ao usuário selecionar a avaliação para o qual irá realizar o lançamento de notas;
RF06	o sistema deverá permitir ao usuário selecionar a forma de lançamento de notas.
RF07	o sistema deverá permitir ao usuário navegar entre os alunos, lançar as notas e realizar a gravação das mesmas utilizando comandos de voz.
RF08	o sistema deverá permitir ao usuário consultar as notas lançadas.

Fonte: elaborado pelo autor.

Quadro 4 - Principais requisitos não funcionais do protótipo

Requisitos Não Funcionais (RNF)	
RNF01	o sistema será desenvolvido na plataforma de desenvolvimento Java (versão 7).
RNF02	o sistema utilizará o banco de dados MSSQL (versão 2008 ou superior).
RNF03	o sistema utilizará a biblioteca Web Speech API para reconhecimento de voz.
RNF04	o sistema será totalmente web, necessitando de internet para ser acessado.
RNF05	o sistema somente estará homologado para o browser Chrome versão 25 ou superior.
RNF06	o sistema necessita de hardware que permita a entrada de som (microfone) para realizar o lançamento de notas utilizando a voz.

Fonte: elaborado pelo autor.

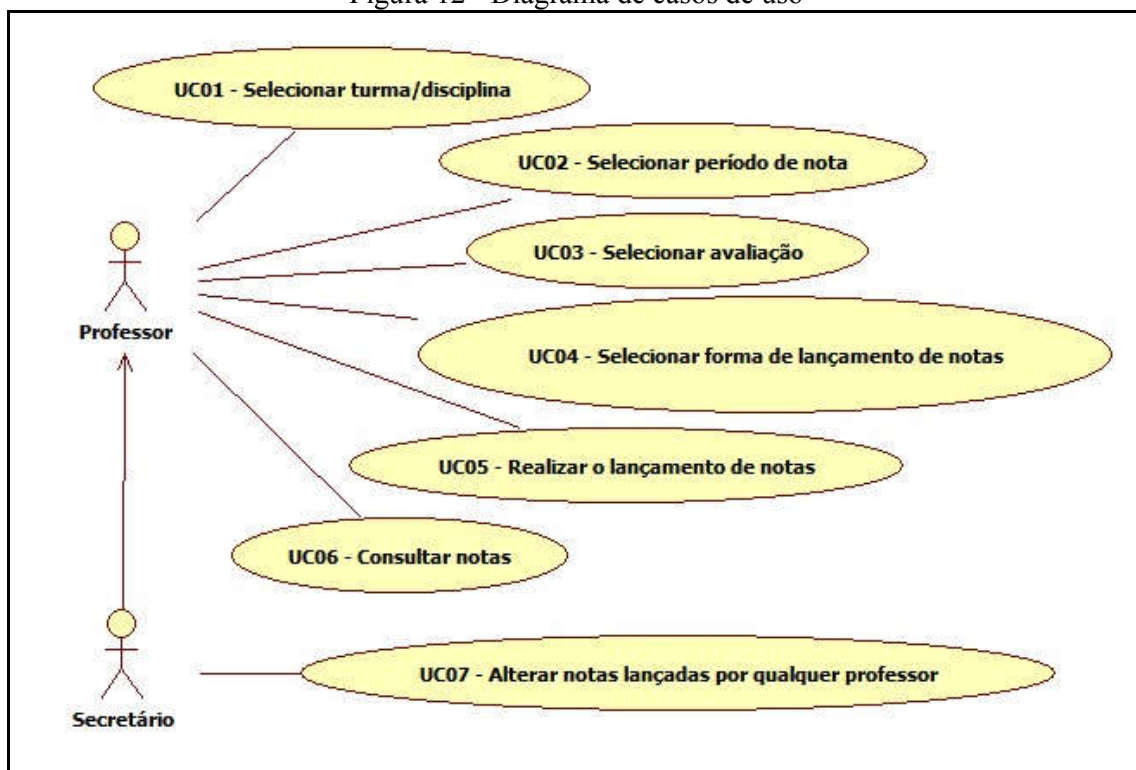
3.2 ESPECIFICAÇÃO

Na especificação do trabalho foi utilizada a ferramenta StartUML versão 5.0.2. Foram criados os diagramas de casos de uso e diagrama de classes.

3.2.1 Diagrama de casos de uso

Nesta seção apresentam-se os casos de uso do protótipo, conforme ilustra a Figura 12. Identificou-se os atores, denominados *Professor* e *Secretário*, os quais utilizam todas as funcionalidades do protótipo.

Figura 12 - Diagrama de casos de uso



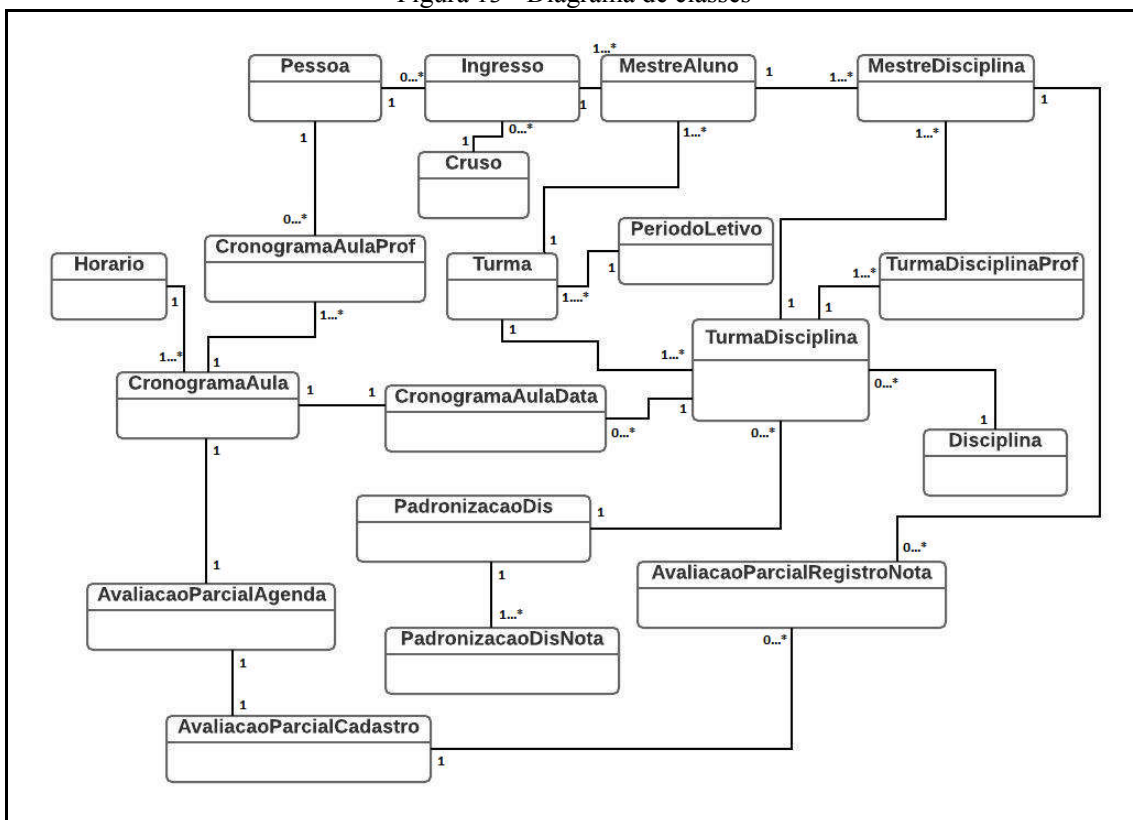
Fonte: elaborado pelo autor.

O ator denominado *Professor* é responsável por realizar o processo de lançamento das notas dos alunos, com a possibilidade de escolher entre lançamento manual ou lançamento por voz. Já o ator denominado *Secretário*, possui a mesma responsabilidade do professor, porém, com a possibilidade de alterar as informações registradas por qualquer docente.

3.2.2 Diagrama de classe

Esta seção apresenta o diagrama de classe do protótipo, contendo o relacionamento entre as classes. A Figura 13 ilustra as classes utilizadas pelo protótipo.

Figura 13 - Diagrama de classes



Fonte: elaborado pelo autor.

As classes ilustradas na Figura 13 são utilizadas para controlar o cadastro e manutenção das informações que serão registradas pelo usuário, sendo elas:

- a) classe *Pessoa* – classe responsável por armazenar o cadastro de pessoa;
- b) classe *Curso* – classe responsável por armazenar o cadastro de curso;
- c) classe *Ingresso* – representa o ingresso de uma pessoa em um determinado curso;
- d) classe *PeríodoLetivo* – classe que representa a estrutura do período letivo;
- e) classe *Turma* – representa o cadastro de turma de um determinado período letivo;
- f) classe *MestreAluno* – representa a matrícula de um aluno em uma determinada turma;
- g) classe *Disciplina* – representa a estrutura do cadastro de disciplina;
- h) classe *TurmaDisciplina* – armazena as disciplinas de uma turma;
- i) classe *MestreDisciplina* – representa a matrícula de um aluno em uma turma/disciplina;
- j) classe *Horario* – classe responsável por armazenar o horário de aula;
- k) classe *CronogramaAula* – representa a estrutura do cadastro de cronograma de aula, de um determinado horário;

- l) classe `CronogramaAulaData` – armazena a data da aula de uma determinada turma/disciplina;
- m) classe `CronogramaAulaProf` – representa o vínculo do professor que está lecionando a turma/disciplina em uma determinada data;
- n) classe `TurmaDisciplinaProf` – vínculo direto entre o professor e uma turma/disciplina.
- o) classe `PadronizacaoDis` – representa o critério de avaliação que poderá ser utilizado por uma determinada turma/disciplina;
- p) classe `PadronizacaoDisNota` – armazena os tipos de nota de um determinado critério de avaliação (ex.: 1º bimestre, 2º bimestre...);
- q) classe `AvaliacaoParcialCadastro` – representa o cadastro de avaliação parcial;
- r) classe `AvaliacaoParcialAgenda` – representa o agendamento de uma determinada avaliação;
- s) classe `AvaliacaoParcialRegistroNota` – classe que armazena a nota que o aluno recebeu em uma determinada avaliação.

A principal classe do protótipo é a `AvaliacaoParcialRegistroNota`, que armazena a nota que foi registrada para determinado aluno na respectiva avaliação. Os atributos fundamentais das principais classes do protótipo estão descritos no Apêndice C.

3.3 IMPLEMENTAÇÃO

Nesta seção evidenciam-se as técnicas e ferramentas utilizadas na implementação do protótipo, a descrição do desenvolvimento do protótipo e a sua operacionalidade.

3.3.1 Técnicas e ferramentas utilizadas

O protótipo foi desenvolvido utilizando o ambiente de desenvolvimento Eclipse Java EE IDE versão 4.5.2, a linguagem Java versão 7.0 e o *framework* PrimeFaces versão 6.0. Também foi utilizada a biblioteca Web Speech API, já integrada ao navegador Chrome a partir da versão 25, para realizar o reconhecimento de voz. As informações registradas são armazenadas em banco de dados MSSQL versão 2008 ou superior. Para encontrar os alunos no sistema, foi utilizada a técnica do algoritmo de Levenshtein, que visa comparar palavras em busca do termo mais próximo do desejado.

3.3.2 Etapas para realizar o reconhecimento de voz

O reconhecimento da fala do usuário é realizado em quatro etapas:

- a) inicializar a biblioteca de reconhecimento de voz;
- b) receber o resultado final retornado pela API;
- c) identificar palavras chaves e interpretar o resultado obtido;
- d) realizar as operações solicitadas por comando de voz, no protótipo.

Para inicializar o reconhecimento de voz, instancia-se um objeto da classe `SpeechRecognition`, alterando o valor da propriedade `continuous` para `true`, desta forma, tornando o reconhecimento de voz contínuo. Logo após, inicia-se o reconhecimento, invocando o método `start`. O Quadro 5 apresenta o código Javascript utilizado para inicializar a biblioteca de reconhecimento de voz.

Quadro 5 - Inicializando o reconhecimento de fala

1	<code>//instancia o serviço de reconhecimento de fala</code>
2	<code>var recognizer = new window.SpeechRecognition();</code>
3	
4	<code>//reconhecimento contínuo (retorno ao final)</code>
5	<code>recognizer.continuous = true;</code>
6	
7	<code>//inicializa o reconhecimento</code>
8	<code>recognizer.start();</code>

Fonte: elaborado pelo autor.

Uma vez que o reconhecimento foi inicializado, o resultado final é obtido no momento em que houver pausa na fala do usuário. A API disponibiliza vários eventos que possibilitam realizar a manipulação dos resultados, porém, para o desenvolvimento do protótipo foi utilizado o evento `onresult`. Esse evento é chamado ao constatar-se uma pausa na fala do usuário, retornando uma lista com os resultados obtidos. Um dos objetos desta lista representa o resultado final, identificado pelo atributo booleano `isFinal`, conforme ilustra o Quadro 6.

Quadro 6 - Identificando o resultado final obtido

1	//evento responsável por receber o resultado final do reconhecimento
2	recognizer.onresult = function(event) {
3	//variável que irá armazenar o texto reconhecido
4	var textoFalado = "";
5	
6	//percorre os resultados em busca do resultado final
7	for (var i = event.resultIndex; i < event.results.length; i++) {
8	//verifica se é o resultado final
9	if(event.results[i].isFinal){
10	//atribui o resultado para a variável
11	textoFalado = event.results[i][0].transcript;
12	textoFalado = textoFalado.toUpperCase().trim();
13	
14	//método responsável por interpretar palavras chaves
15	realizaReconhecimento(textoFalado);
16	}
17	}
18	}

Fonte: elaborado pelo autor.

O método `realizaReconhecimento` executado na linha quinze do Quadro 6 é responsável por identificar se o usuário mencionou uma das palavras chaves disponibilizadas pelo protótipo. O Quadro 7 ilustra as palavras chaves que poderão ser utilizadas para executar as ações no sistema.

Quadro 7 - Palavras chaves aplicáveis

Palavra chave	Ação executada	Exemplo da fala
Período	deverá ser utilizada para selecionar o período de avaliação.	período primeiro bimestre
Avaliação	deverá ser utilizada para selecionar a avaliação para qual será realizado o lançamento da nota.	avaliação primeira avaliação
Aluno/Aluna	deverá ser utilizada precedendo o nome do aluno. O protótipo irá identificar o aluno na lista e colocar o foco automaticamente no campo de nota referente a ele.	aluno João da Silva
Nota	deverá ser utilizada ao estar com o foco no campo de nota do respectivo aluno, precedendo a nota que será informada.	nota 9.35
Ação gravar	deverá ser utilizada para executar o comando que realiza a gravação dos registros de notas lançados.	ação gravar
Próximo	deverá ser utilizada ao estar com o foco em algum campo de entrada de dados, para focar no próximo campo disponível.	próximo
Anterior	deverá ser utilizada ao estar com o foco em algum campo de entrada de dados, para focar no campo anterior disponível.	anterior
Limpar	deverá ser utilizada ao estar com o foco em um dos campos de nota, para limpar o valor informado anteriormente.	limpar
Cancelar	deverá ser utilizada para cancelar a ação da chamada de algum aluno. Exemplo de utilização: está focado em algum campo de nota, e falou a palavra “aluno”, porém, ainda quer lançar a nota para o aluno que está atualmente com o foco. Então, pode-se mencionar a palavra chave “cancelar”, permitindo assim, lançar a nota para o aluno focado.	cancelar

Fonte: elaborado pelo autor.

O método `realizaReconhecimento` é fundamental ao funcionamento do protótipo. O Quadro 8 ilustra a primeira parte do método, que identifica a palavra chave ou o valor que sucede esta palavra.

Quadro 8 - Identificando a palavra chave ou valor que a sucede (método: `realizaReconhecimento`)

```
1 textoFalado = textoFalado.trim();
2 var primeiraPalavra = "";
3
4 //se contém espaço, armazena a primeira palavra
5 var contemEspaco = false;
6 if(textoFalado.indexOf(" ") > 0){
7     primeiraPalavra = textoFalado.split(" ")[0];
8     contemEspaco = true;
9 }else{//se não contém espaço, a primeira palavra é o que foi falado
10     primeiraPalavra = textoFalado;
11 }
```

Fonte: elaborado pelo autor.

Quando o texto falado não contiver espaço, o protótipo identifica se foi mencionada uma palavra chave. Caso afirmativo, o sistema verifica qual foi essa palavra, e em seguida armazena o valor “verdadeiro” em um atributo de controle booleano, respectivo a essa chave. Este atributo será utilizado posteriormente quando o usuário falar a próxima palavra, que deverá ser o valor sucessor desta chave. Por exemplo: o usuário menciona a palavra “aluno”. No momento de pausa, o resultado retornado é “aluno”. Nesse caso, o protótipo armazena no atributo `ultimaPalavraAluno` o valor “verdadeiro”. Após a pausa, o usuário menciona o nome do aluno. Posteriormente, o protótipo verifica se o atributo `ultimaPalavraAluno` está com valor “verdadeiro”. Em caso afirmativo, invoca-se o método do servidor `encontraRegistroLista`, que irá identificar o respectivo aluno na lista. O Quadro 9 ilustra o trecho de código responsável por realizar este processamento.

Quadro 9 - Processando palavra chave, com pausa (método: realizaReconhecimento)

```

1 //Se não contiver espaço
2 if(!contemEspaco || ultimaPalavraAluno){
3     //verifica se falou a palavra "cancelar"
4     if(isFalouCancelar(primeiraPalavra)){
5         ultimaPalavraAluno = false;
6         ultimaPalavraNota = false;
7         ultimaPalavraAcao = false;
8         return false;
9     }else if(isFalouAcao(primeiraPalavra)){//se falou ação
10        ultimaPalavraAcao = true;
11        return false;
12    }else if(isFalouAluno(primeiraPalavra)){//falou a palavra ALUNO
13        ultimaPalavraAluno = true;
14        return false;
15    }else if(isFalouNota(primeiraPalavra)){//falou a palavra NOTA
16        ultimaPalavraNota = true;
17        return false;
18    }
19
20    if(ultimaPalavraAluno){//chama o servidor pra achar o aluno
21        encontraRegistroLista ([{name:"paramNomeAlunoProcurar",
value: textoFalado}]);
22    }else if(ultimaPalavraNota){//chama o servidor pra lançar nota
23        encontraRegistroLista ([{name:"paramNomeAlunoProcurar",
value: 'NOTA ' + textoFalado}]);
24    }else if(ultimaPalavraAcao && isFalouGravar(textoFalado)){
25        //clica no botão Gravar
26        $("#formPrincipal\\:btnGravar").click();
27    }else if(isFalouLimpar(textoFalado)){
28        $(':focus').val("");
29    }else if(isFalouProximo(textoFalado)){
30        //aqui joga o foco pro próximo input
31    }else{//emite feedback pois não falou nenhuma das chaves
32        setMsgErro('Não entendi! Tente novamente.');
```

33 return false;

34 }

35 }

Fonte: elaborado pelo autor.

Quando o texto mencionado contiver espaço, o protótipo identifica se a primeira palavra da frase é uma palavra chave. Se não for, é emitido um feedback para o usuário. Caso contrário, verifica-se qual foi a palavra chave mencionada, e logo em seguida, realiza-se a execução do processo respectivo, conforme ilustra o Quadro 10.

Quadro 10 - Processando palavra chave, sem pausa (método: realizaReconhecimento)

```

1 //quebra o texto falado por espaço, pra identificar a chave
2 var textoFaladoSplit = textoFalado.split(" ");
3
4 //textoFaladoSplitP1 => é a chave
5 var textoFaladoSplitP1 = textoFaladoSplit[0];
6 //valor que sucede a palavra chave
7 var textoFaladoSplitP2 = textoFaladoSplit[1];
8 //retira o espaço em excesso
9 textoFaladoSplitP1 = textoFaladoSplitP1.trim();
10 textoFaladoSplitP2 = textoFaladoSplitP2.trim();
11 //se a primeira palavra não for palavra chave, emite feedback
12 if(!isFalouAluno(textoFaladoSplitP1) &&
!isFalouNota(textoFaladoSplitP1) && !isFalouAcao(textoFaladoSplitP1)
&& !isFalouLimpar(textoFaladoSplitP2)){
13     setMsgErro('Não entendi! Tente novamente');
14 }else{
15     //se não for a palavra "Ação",
16     //chama o servidor pra encontrar o aluno e lançar a nota
17     if(!isFalouAcao(textoFaladoSplitP1)){
18         encontraRegistroLista([{name:"paramNomeAlunoProcurar",
value: textoFalado}]);
19     }else if(isFalouAcao(textoFaladoSplitP1) &&
20 isFalouGravar(textoFaladoSplitP2)){
21         //se falou "Ação gravar", clica no botão de gravar
22         $("#formPrincipal\\:btnGravar").click();
23     }else if(isFalouLimpar(textoFaladoSplitP2)){
24         $(':focus').val("");
25     }else{//se chegou aqui, não identificou o texto falado
26         setMsgErro('Não entendi!');
27         return false;
28     }
29 }

```

Fonte: elaborado pelo autor.

Para identificar o aluno, ou realizar o lançamento da nota, o processamento é realizado no servidor, em código Java. No Quadro 10, linha dezoito, há um exemplo da chamada do método `encontraRegistroLista` para o servidor. Esse método irá identificar se o texto falado foi o nome do aluno ou o valor da nota, levando em consideração de que somente é chamado quando os precedentes foram cumpridos. No caso do nome do aluno, é utilizado o algoritmo de Levenshtein para encontrar o nome mais próximo possível ao que foi falado.

3.3.3 Implementação do Algoritmo de Levenshtein

Para a implementação do algoritmo de Levenshtein, foram testados dois métodos equivalentes, desenvolvidos em linguagens diferentes: um em Javascript (executado do lado do cliente) e outro em Java (executado do lado do servidor). Na realização dos testes, foi utilizada uma lista com 400 palavras exatamente iguais para a execução de ambos os códigos. Identificou-se que o desempenho do código em Javascript teve inúmeras variações, entre 0,11 e 0,27 milissegundos. Já o código em Java, se manteve mais estável, entre 0,15 e 0,16

milissegundos. Portanto, optou-se por utilizar o algoritmo de Levenshtein desenvolvido na linguagem Java, conforme ilustra o Quadro 11.

Quadro 11 - Algoritmo de Levenshtein em Java

```

1 public static int levenshtein(String textoFalado, String
  nomeAlunoLista) {
2     textoFalado = textoFalado.toLowerCase();
3     nomeAlunoLista = nomeAlunoLista.toLowerCase();
4     int[] matriz = new int[nomeAlunoLista.length() + 1];
5     for (int j = 0; j < matriz.length; j++)
6         matriz[j] = j;
7     for (int i = 1; i <= textoFalado.length(); i++) {
8         matriz[0] = i;
9         int nw = i - 1;
10        for (int j = 1; j <= nomeAlunoLista.length(); j++) {
11            int cj = Math.min(1 + Math.min(matriz[j], matriz[j
- 1]), textoFalado.charAt(i - 1) == nomeAlunoLista.charAt(j - 1) ? nw
: nw + 1);
12                nw = matriz[j];
13                matriz[j] = cj;
14        }
15    }
16    return matriz[nomeAlunoLista.length()];
17 }

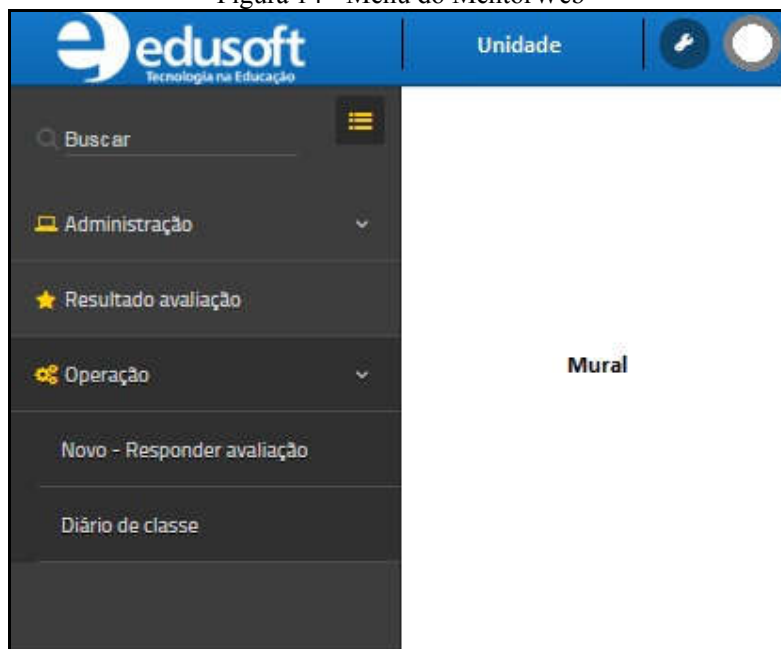
```

Fonte: adaptado de Bacco (2014, p. 1).

3.3.4 Operacionalidade do protótipo

Esta seção apresenta o funcionamento do protótipo. O protótipo é acessado através do sistema de gestão acadêmica MentorWeb, menu Diário de classe, conforme ilustra a Figura 14.

Figura 14 - Menu do MentorWeb



Fonte: elaborado pelo autor.

Após acessar o item de menu Diário de classe, apresenta-se a tela para seleção da turma/disciplina. Essa tela permite realizar a pesquisa informando os filtros de período letivo, curso, turma, disciplina e situação da turma. Na central do professor, a pesquisa retornará somente as turmas/disciplinas na qual o professor está vinculado. Já para a secretaria, listará todas as informações disponíveis. A Figura 15 ilustra a tela de seleção de turma/disciplina na central do professor (a versão *mobile* da tela encontra-se no Apêndice B).

Figura 15 - Tela de seleção de turma/disciplina

Turma / Disciplina	Curso	Notas e faltas
ADM_1 / Filosofia e Ética	Graduação em Administração (Noturno) Bacharelado	
ADM_1 / Psicologia	Graduação em Administração (Noturno) Bacharelado	

Fonte: elaborado pelo autor.

Uma das colunas retornadas na tela da Figura 15 é a coluna Notas e faltas, que apresenta um ícone. Ao clicar neste ícone, o usuário é redirecionado para a tela de lançamento/consulta de notas das avaliações, conforme mostra a Figura 16 (a versão *mobile* da tela encontra-se no Apêndice B).

Figura 16 - Tela para lançamento/consulta de notas

Alunos	Notas
Aporele Mariane Zaia	5,00
Eduardo Marques Ferreira	5,00
Fellippe Beckenbauer	3,00
João Thiago Ferreira dos Santos	3,99
Lucas Goulart Foster	5,99
Maria Helena da Rosa	3,99
Maria Helena Sonogo	2,95

Fonte: elaborado pelo autor.

A tela ilustrada na Figura 16 apresenta os filtros de Período e Avaliação. O filtro Avaliação é sucessivo ao filtro de Período, isso significa que ao selecionar um período, o filtro de avaliação somente listará as avaliações aplicadas no período selecionado. Após selecionar a avaliação, o protótipo carrega abaixo as notas dos alunos, caso já tenham sido lançadas. Ao carregar a tela o primeiro período já vem como padrão selecionado, bem como a primeira avaliação. Ao selecionar um período, a primeira avaliação já é automaticamente selecionada. Ao lado do botão Gravar há o ícone `microfone`, que habilita/desabilita o recurso de comando de voz. Quando o recurso estiver habilitado, o `microfone` fica piscando, já quando estiver desabilitado, permanece cinza.

3.4 RESULTADOS E DISCUSSÕES

Esta seção apresenta quatro experimentos para o protótipo. O primeiro para validar a busca dos alunos a partir do nome, o segundo para validar o processo de lançamento das notas como um todo e o terceiro para validar o processo de lançamento de notas por voz em um dispositivo móvel. Por fim, o quarto experimento visa verificar a usabilidade do protótipo.

3.4.1 Experimento 01: validar a busca dos alunos a partir do nome

O objetivo deste experimento é validar a busca dos alunos no protótipo a partir da menção do seu nome, tanto pelo nome completo, bem como primeiro nome. Para este experimento, foram utilizados dez alunos. O Quadro 12 ilustra o nome de cada aluno, o que foi reconhecido pela API, e se encontrou o aluno correto após aplicar a técnica de Levenshtein.

Quadro 12 - Experimento 01: validação da busca dos alunos a partir do nome

Nome do aluno	Retorno da API (1° nome)	Retorno da API (completo)	Sucesso
Alexander Julius Oliveira	Alexandre	Alexandre Júlio Oliveira	Sim
Aporele Mariane Zaia	Aporelly	Aporelly Mariane Saia	Sim
Daniel Augusto da Silva	Daniel	Daniel Augusto da Silva	Sim
Eduardo Marques Ferreira	Eduardo	Eduardo Marques Ferreira	Sim
Fellippe Beckenbauer	Felipe	Felipe Beckenbauer	Sim
João Thiago Ferreira dos Santos	João	João Tiago Ferreira dos Santos	Sim
Lucas Goulart Foster	Lucas	Lucas Goulart Foster	Sim
Maria Helena da Rosa	--	Maria Helena da Rosa	Sim
Maria Helena Sonego	--	Maria Helena Sonego	Sim
Roberto Beckenbauer	Roberto	Roberto Beckenbauer	Sim

Fonte: elaborado pelo autor.

A partir do Quadro 12, pode-se perceber que se obteve sucesso em 100% dos casos de teste após aplicar a técnica de Levenshtein. Porém, quando tem-se dois alunos com o mesmo nome, não é ideal mencionar somente o primeiro nome, pois nesse caso o cursor será

posicionado sempre no primeiro aluno encontrado. Quando há casos com nomes que possuem grafias diferentes, porém com a mesma pronúncia, o ideal é mencionar o nome completo do aluno. Analisando o nome “Fellipe”, pode-se observar que o retorno da API foi “Felipe”. Se houvesse um aluno com o nome “Felipe” na lista, este seria selecionado. O resultado obtido a partir deste experimento foi satisfatório, atendendo as expectativas.

3.4.2 Experimento 02: validar o lançamento de notas

O objetivo deste experimento é validar o processo de lançamento de notas, explorando as palavras chaves e seu respectivo valor. O Quadro 13 apresenta a palavra chave mencionada, o respectivo valor e a ocorrência de sucesso ou erro.

Quadro 13 - Experimento 02: validação do lançamento de notas

Palavra chave	Valor	Sucesso	Erro
Período	Segundo semestre	Sim	
Avaliação	Quinta avaliação	Sim	
Aluno	Eduardo Marques	Sim	
Nota	9.83	Sim	
Limpar	--	Sim	
Nota	7,5	Sim	
Próximo	--	Sim	
Nota	8.28	Sim	
Aluno	Lucas	Sim	
Nota	8.38	Não	API retornou “8 PONTO 38”, e o protótipo inseriu a parte numérica “8.00”.
Nota	8,38	Sim	
Ação	Gravar	Sim	
Avaliação	Quarta avaliação	Sim	
Aluno	Roberto Beckenbauer	Sim	
Nota	10	Sim	
Anterior	--	Sim	
Nota	9,75	Sim	
Limpar	--	Sim	
Nota	9.5	Sim	
Ação	Gravar	Sim	

Fonte: elaborado pelo autor.

A partir do Quadro 13, percebe-se que o protótipo teve um aproveitamento de 95% na execução das ações por comandos de voz. Pelos testes realizados, a maioria das notas informadas foram interpretadas corretamente pela API. Entretanto, em casos específicos, como no caso da nota “8.38”, a API interpretou como um valor texto e não numérico. Quando isto ocorrer, pode-se trocar a palavra “ponto” por “vírgula”. Neste caso, a nota é interpretada corretamente como um valor numérico. Quando o comando não é identificado pelo protótipo, é emitido um feedback em tela. O resultado obtido a partir deste experimento foi satisfatório.

3.4.3 Experimento 03: validar o lançamento de notas em um dispositivo móvel

O objetivo deste experimento é validar o processo de lançamento de notas a partir de um dispositivo móvel. Para realizar a validação foi utilizado o smartphone Samsung Galaxy S3 mini, e os mesmos dados ilustrados no Quadro 13.

O resultado do teste não foi totalmente como esperado. Identificou-se que o navegador Google Chrome, em dispositivos móveis, ignora a propriedade `continuous` da API de reconhecimento de voz. Com isso, a cada frase mencionada o evento `onend` é disparado, fazendo com que o reconhecimento seja parado e iniciado novamente. Além disso, a cada vez que o reconhecimento é reiniciado, o próprio navegador emite um som de aviso, e somente após esse som, é possível mencionar o comando desejado. Caso a tela do dispositivo desligue, é necessário clicar no ícone `microfone` para reiniciar a API. Contudo, mencionando os comandos após o aviso do navegador, obteve-se 95% de assertividade no lançamento de notas, equiparando-se ao resultado do experimento 02.

3.4.4 Experimento 04: teste de usabilidade

O objetivo deste experimento é analisar a usabilidade do protótipo. Para isso, foi aplicado um questionário referente ao teste de usabilidade (contido no Apêndice D) com seis pessoas.

O experimento foi realizado individualmente, utilizando um notebook Sony Vaio, portando o sistema operacional Windows 7, para a realização do teste em computador. Para a realização do teste em dispositivo móvel, foi utilizado o smartphone Samsung Galaxy Note, portando o sistema operacional Android. A lista de instruções para o usuário realizar o teste foi anexada ao questionário.

Na primeira fase, é realizada a análise do perfil do usuário. Foram envolvidos seis usuários no teste de usabilidade, tanto com experiência em qualidade de software, bem como em desenvolvimento. O Quadro 14 apresenta o perfil destes usuários.

Quadro 14 - Perfil de usuários que realizaram o teste de usabilidade

Sexo	100% masculino
1. Idade	66,7% entre 18 e 30 anos 16,65% entre 30 e 40 anos 16,65% mais de 40 anos
2. Grau de conhecimento do MentorWeb	50% alto 33,33% médio 16,67% baixo
3. Já utilizou a digitação de notas atual	50% sim 50% não

Fonte: elaborado pelo autor.

A partir do Quadro 14, pode-se perceber que metade dos usuários envolvidos no teste possuem alto conhecimento do sistema atual (MentorWeb) e/ou já utilizaram a digitação de notas nesse sistema. O Quadro 15 apresenta os resultados obtidos através do questionário de usabilidade aplicado, em ambos os dispositivos disponibilizados.

Quadro 15 - Respostas referentes ao questionário de usabilidade

Pergunta	Notebook	Smartphone
1. Na sua opinião, o protótipo ficou intuitivo e fácil de utilizar?	100% Sim	100% Sim
2. Os comandos de voz foram reconhecidos com sucesso?	66,7% Sim 33,3% A maioria	100% Sim
3. Quando o comando de voz não é reconhecido, ou é inválido, o protótipo está emitindo feedback?	100% Sim	100% Sim
4. Em uma escala de 0 a 10, qual a sua nota para o protótipo?	Média: 9,50	Média: 9,35
5. Você teve alguma dificuldade ao utilizar o protótipo?	16,6% Sim 83,4% Não	33,3% Sim 66,7% Não

Fonte: elaborado pelo autor.

Analisando o Quadro 15 pode-se observar que todos os usuários envolvidos no teste de usabilidade acharam o protótipo intuitivo e fácil de utilizar. Entretanto, pode-se perceber que alguns deles tiveram dificuldade ao utilizar o protótipo. Quando indagado o motivo desta dificuldade, foram citados os seguintes itens:

- a) no smartphone, quando o protótipo emite um feedback por não reconhecer um comando, após informar o nome de um aluno o foco é posicionado corretamente na caixa de texto correspondente, porém, precisa rolar a tela pra visualizar;
- b) no notebook, quando há muitas conversas paralelas próximas ao microfone, as vezes necessita-se repetir os comandos para serem reconhecidos corretamente. Porém, com o uso de um headphone esta interferência é menor.

4 CONCLUSÕES

O objetivo principal do trabalho, que consistiu em realizar a construção de um protótipo para o lançamento de notas acadêmicas no MentorWeb utilizando reconhecimento de voz, foi alcançado. O primeiro objetivo específico era capturar o áudio a partir de um microfone. O objetivo foi atingido, porém, em testes realizados, percebeu-se que ao utilizar o protótipo pelo computador, o ideal é que se utilize um headphone para captar o áudio, por ter eficácia melhor no reconhecimento. O segundo objetivo era integrar a biblioteca de reconhecimento de voz com o protótipo. O objetivo foi alcançado utilizando a biblioteca Web Speech API, comprovando por testes de usabilidade que é possível realizar o reconhecimento de voz, possibilitando o lançamento de notas no protótipo. O terceiro objetivo específico era utilizar uma técnica de comparação de palavras para encontrar os alunos na lista. Neste caso, foi empregado o algoritmo de Levenshtein, que visa comparar as palavras em busca da sentença mais próxima ao que foi mencionado.

O uso da biblioteca Web Speech API não garante que a interpretação das sentenças mencionadas pelo usuário seja perfeita. Porém, pelos testes realizados, incluindo teste de usabilidade, a maioria dos comandos mencionados foram interpretados com sucesso. Verificou-se que em dispositivos móveis a biblioteca de reconhecimento de voz não consegue interpretar a propriedade `continuous`, fazendo com que seja reiniciada automaticamente a cada intervalo de tempo.

A partir dos testes realizados com os usuários, constatou-se que o protótipo ficou intuitivo e simples de utilizar, independente do dispositivo utilizado. O protótipo atingiu o resultado esperado na maioria dos casos de testes, entretanto, é válida a realização de estudos mais profundos, a fim de melhorar ainda mais o reconhecimento de voz.

4.1 EXTENSÕES

Pode-se destacar as possíveis extensões abaixo:

- a) possibilitar a utilização do protótipo em outros navegadores;
- b) melhorar o algoritmo de análise das palavras chave, a fim de deixar o reconhecimento ainda mais ágil;
- c) encontrar uma forma de que em dispositivos móveis o reconhecimento de voz não reinicie a cada intervalo de tempo.

REFERÊNCIAS

- ALENCAR, Vladimir F. S. de; ALCAIM, Abraham. **Atributos eficientes em reconhecimento automático de voz distribuído**. [S.I.], 2008. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-17592008000200004>. Acesso em: 10 nov. 2016.
- BALSINI, Luis F. **SapiSim: Um módulo para identificação de coautorias nas produções científicas do SAPI**. 2012. 110 f. Trabalho de Conclusão de Curso (Técnico-científico em Ciências da Computação) – Centro de Ciências Tecnológicas da Terra e do Mar, Universidade do Vale do Itajaí, Itajaí.
- DAUGS, Carla J. **O que é um site responsivo?** [S.I.], 2016. Disponível em: <http://www.valedaweb.com.br/_265-o-que-e-um-site-responsivo>. Acesso em: 02 set. 2016.
- EDUSOFT. **Histórico das Versões/Releases do Mentor Web**. [S.I.], [2012?]. Disponível em: <<http://help.edusoft.com.br/MentorWEB/releaseNotes/HTML/index.html>>. Acesso em: 19 nov. 2016.
- EDUSOFT. **Escolhendo um sistema de gestão escolar**. Blumenau, 2014a. Disponível em: <<http://site.edusoft.com.br/noticias/id/39>>. Acesso em: 12 set. 2015.
- EDUSOFT. **Escolhendo um sistema de gestão escolar**. Blumenau, 2014b. Disponível em: <<http://site.edusoft.com.br/textos/missao-visao>>. Acesso em: 12 set. 2015.
- GUERRATO, Dani. **Design Responsivo na prática 2: do layout ao HTML**. [S.I.], 2014. Disponível em: <<http://tableless.com.br/design-responsivo-na-pratica-2-layout-ao-html>>. Acesso em: 15 jul. 2016.
- DAUGS, Carla J. **O que é um site responsivo?** [S.I.], 2016. Disponível em: <http://www.valedaweb.com.br/_265-o-que-e-um-site-responsivo>. Acesso em: 02 set. 2016.
- KLERING, Katiele. **O crescimento de acessos à internet por dispositivos móveis**. [S.I.], 2015. Disponível em: <<https://www.imobex.com.br/comunidade/o-crescimento-de-acessos-a-internet-por-dispositivos-moveis/#.WDYjrtUrK1s>>. Acesso em: 04 nov. 2016.
- MORAN, José M. **Gestão inovadora da escola com tecnologias**. São Paulo, 2003. Disponível em: <http://www.eca.usp.br/prof/moran/site/textos/tecnologias_eduacao/gestao.pdf>. Acesso em: 04 set. 2015.
- MOURA, Amauri. **Design responsivo: o que é e por que você precisa saber**. [S.I.], 2016. Disponível em: <<https://moip.com.br/blog/design-responsivo-por-que-voce-precisa-saber>>. Acesso em: 05 set. 2016.
- OLIVEIRA, Neilza A. de. **Reconhecimento de fala utilizando Modelos Matemáticos e Redes Neurais**. 2002. 54 f. Dissertação (Mestrado em Ciências da Computação) - Programa de Pós-graduação em Ciências da Computação, Universidade Federal de Santa Catarina, Florianópolis.
- OLIVEIRA, Paulo H. A. de. **Desenvolvimento de uma plataforma para um jogo de xadrez inteligente controlado por reconhecimento de voz**. 2014. 77 f. Trabalho de Conclusão de Curso (Curso de Engenharia Elétrica com ênfase em eletrônica) - Departamento de Engenharia Elétrica e Computação, Universidade de São Paulo, São Carlos.
- PEREIRA, Ana P. **Como funciona o reconhecimento de voz?**. [S.I.], 2009. Disponível em: <<http://www.tecmundo.com.br/curiosidade/3144-como-funciona-o-reconhecimento-de-voz-.htm>>. Acesso em: 01 set. 2015.

PORVIR. **Tecnologia na Educação:** Recomendações e experiências para transformar a maneira como se ensina e aprende a partir do uso de ferramentas digitais. [São Paulo], 2015. Disponível em: <<http://porvir.org/especiais/tecnologia>>. Acesso em: 18 set. 2015.

SHIRES, Glen.; WENNBORG, Hans. **Web Speech API.** [S.I.], 2012. Disponível em: <<https://dvc.w3.org/hg/speech-api/raw-file/tip/speechapi.html>>. Acesso em: 10 set. 2015.

SILVA, Maria E. V. da. **XSimilarity:** Uma ferramenta para consultas por similaridade embutidas na linguagem XQuery. 2007. 47 f. Trabalho de Graduação - Universidade Federal do Rio Grande do Sul, Porto Alegre.

SOUZA, Charles R. **Edusoft - Solicitação por funcionalidades mobile.** [mensagem pessoal]. Mensagem recebida por <adrian.possamai@edusoft.com.br> data de recebimento, 20 nov. 2015.

VIEIRA, Marcelo A. **Voz da Notícia:** Aplicativo web de notícias para deficientes visuais. 2014. 64 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) - Departamento de Ciências Exatas e Tecnológicas, Universidade Estadual do Sudoeste da Bahia, Vitória da Conquista.

YNOGUTI, Carlos A. **Reconhecimento de Fala Contínua Usando Modelos Ocultos de Markov.** 1999. 138 f. Tese (Doutorado em Engenharia Elétrica) – Departamento de comunicações, Universidade Estadual de Campinas, Campinas.

APÊNDICE A – Descrição dos Casos de Uso

Este Apêndice apresenta a descrição dos casos de uso conforme previsto no diagrama apresentado na seção 3.2.1, que vão do Quadro 16 até o Quadro 22.

Quadro 16 - UC01 - Selecionar turma/disciplina

UC01 Selecionar turma/disciplina

Caso de uso onde pode-se selecionar a turma/disciplina para qual será realizado o lançamento de notas.

Atores: Professor/Secretário

Constraints:

Pré-condição – O ator deve estar logado no sistema MentorWeb.

Pós-condição – Turma/disciplina selecionada e o protótipo apresenta a lista com os respectivos alunos.

Cenário:

1. O ator acessa o menu diário de classe.
2. O protótipo apresenta os filtros para pesquisa de turmas/disciplinas.
3. O ator informa os filtros desejados e realiza a pesquisa.
4. O protótipo retorna as turmas/disciplinas disponíveis.
5. O ator clica no ícone *Provas e trabalhos* respectivo à turma/disciplina desejada.

Fonte: elaborado pelo autor.

Quadro 17 – UC02 - Selecionar período de nota

UC02 Selecionar período de nota

Caso de uso onde pode-se selecionar o período de nota na qual será realizado o lançamento das notas (ex.: 1º bimestre, 1º semestre...).

Atores: Professor/Secretário

Constraints:

Pré-condição – O ator deve possuir uma turma/disciplina selecionada.

Pós-condição – São listadas as avaliações aplicadas para o respectivo período.

Cenário:

1. O ator clica na combo *Período* e seleciona qual o período de nota desejado.

Fonte: elaborado pelo autor.

Quadro 18 - UC03 - Selecionar avaliação

<p>UC03 Selecionar avaliação</p> <p>Caso de uso onde pode-se selecionar a avaliação na qual será realizado o lançamento das notas.</p> <p>Atores: Professor/Secretário</p> <p>Constraints:</p> <ul style="list-style-type: none">Pré-condição – O ator deve possuir um período de nota selecionado.Pós-condição – Sistema permite realizar o lançamento das notas. <p>Cenário:</p> <ol style="list-style-type: none">1. O ator clica na combo <i>Avaliação</i> e seleciona a avaliação desejada.
--

Fonte: elaborado pelo autor.

Quadro 19 - UC04 - Selecionar forma de lançamento de notas

<p>UC04 Selecionar forma de lançamento de notas</p> <p>Caso de uso onde pode-se selecionar a forma do lançamento das notas, podendo optar por manual ou comando de voz.</p> <p>Atores: Professor/Secretário</p> <p>Constraints:</p> <ul style="list-style-type: none">Pré-condição – O ator deve estar acessando a tela de lançamento de notas.Pós-condição – Sistema permite realizar o lançamento das notas conforme a opção escolhida. <p>Cenário:</p> <ol style="list-style-type: none">1. O ator seleciona a forma de lançamento de notas clicando no ícone <i>microfone</i>, na barra superior do sistema MentorWeb.
--

Fonte: elaborado pelo autor.

Quadro 20 - UC05 Realizar o lançamento de notas

<p>UC05 Realizar o lançamento de notas Caso de uso onde pode-se realizar o lançamento das notas da avaliação.</p> <p>Atores: Professor/Secretário</p> <p>Constraints: Pré-condição – O ator deve ter selecionado o período de nota, uma avaliação e forma de lançamento de notas. Pós-condição – Notas registradas para a avaliação selecionada.</p> <p>Cenário:</p> <ol style="list-style-type: none"> 1. Manualmente <ol style="list-style-type: none"> 1.1. O ator identifica o aluno na lista. 1.2. O ator clica na caixa de texto ao lado do nome do aluno. 1.3. O ator digita a nota obtida pelo aluno. 1.4. O ator repete os passos 1.1 até 1.3 enquanto possuir notas a serem digitadas. 1.5. O ator clica no botão <i>Gravar</i> para gravar os registros. 2. Por voz <ol style="list-style-type: none"> 2.1. O ator menciona a palavra “aluno” e em seguida, o nome do aluno. 2.2. O protótipo identifica o aluno na lista e coloca o foco na caixa de texto respectiva. 2.3. O ator menciona a palavra “nota” e em seguida, a nota obtida pelo aluno. 2.4. O protótipo lança a nota mencionada na caixa de texto respectiva. 2.5. O ator repete os passos 2.1 até 2.4 enquanto possuir notas a serem lançadas. 2.6. O ator menciona a sentença “ação gravar” para gravar os registros.
--

Fonte: elaborado pelo autor.

Quadro 21 - UC06 Consultar notas

<p>UC06 Consultar notas Caso de uso onde pode-se realizar a conta das notas lançadas para as avaliações.</p> <p>Atores: Professor/Secretário</p> <p>Constraints: Pré-condição – O ator deve ter selecionado o período de nota e uma avaliação. Pós-condição – Notas consultadas.</p> <p>Cenário:</p> <ol style="list-style-type: none"> 1. O ator seleciona o período de notas e a avaliação desejada.

Fonte: elaborado pelo autor.

Quadro 22 - UC07 - Alterar notas registradas por qualquer professor

UC07 Alterar notas registradas por qualquer professor

Caso de uso onde pode-se realizar a manutenção das notas da avaliação sem possuir vínculo com a turma.

Atores: Secretário

Constraints:

Pré-condição – O ator deve ter selecionado o período de nota, uma avaliação e forma de lançamento de notas.

Pós-condição – Notas registradas para a avaliação selecionada.

Cenário:

1. Manualmente
 - 1.1. O ator identifica o aluno na lista.
 - 1.2. O ator clica na caixa de texto ao lado do nome do aluno.
 - 1.3. O ator digita a nota obtida pelo aluno.
 - 1.4. O ator repete os passos 1.1 até 1.3 enquanto possuir notas a serem digitadas.
 - 1.5. O ator clica no botão `Gravar` para gravar os registros.
2. Por voz
 - 2.1. O ator menciona a palavra “aluno” e em seguida, o nome do aluno.
 - 2.2. O protótipo identifica o aluno na lista e coloca o foco na caixa de texto respectiva.
 - 2.3. O ator menciona a palavra “nota” e em seguida, a nota obtida pelo aluno.
 - 2.4. O protótipo lança a nota mencionada na caixa de texto respectiva.
 - 2.5. O ator repete os passos 1.1 até 2.4 enquanto possuir notas a serem lançadas.
 - 2.6. O ator menciona a sentença “ação gravar” para gravar os registros.

Fonte: elaborado pelo autor.

APÊNDICE B – Telas do protótipo em versão mobile

Este Apêndice apresenta as telas do protótipo na versão *mobile*. A Figura 17 ilustra a tela de seleção de turma/disciplina, que é a mesma tela apresentada na Figura 15, porém, em versão *mobile*. A Figura 18 apresenta a tela para lançamento de notas, que é a mesma ilustrada na Figura 16, entretanto, em versão *mobile*.

Figura 17 - Seleção de turma/disciplina

The screenshot shows a mobile application interface for selecting a class/discipline. The interface includes a search bar, a course dropdown menu, and a table with one record. The course dropdown menu is currently set to "Ciências Econômicas - Bacharelado (ECN)". The table has three columns: "Turma / Disciplina", "Curso", and "Notas e faltas". The table contains one record with the following data:

Turma / Disciplina	Curso	Notas e faltas
ECN081-8 / Programação e Projetos II	Ciências Econômicas - Bacharelado	

Registros: 1 de 1.

Fonte: elaborado pelo autor.

Figura 18 - Lançamento de notas

189.45.201.62/MentorWeb, 17:39

edusoft
Tecnologia na Educação

Buscar

Lançamento de notas

Gravar

2016/1 > ECN - Ciências Econômicas - Bacharelado > ECN081-8 > 81PROP2 - Programação e Projetos II

Período
Primeiro Semestre

Avaliação
AV1 - Primeira Avaliação

Alunos	Notas
Aporele Mariane Zaia	5.00
Eduardo Marques Ferreira	5.00
Fellippe Beckenbauer	3.00
João Thiago Ferreira dos Santos	3.99
Lucas Goulart Foster	5.99
Maria Helena da Rosa	3.99
Maria Helena Sonogo	2.95

Fonte: elaborado pelo autor.

APÊNDICE C – Atributos fundamentais das principais classes do protótipo

Este Apêndice apresenta os atributos fundamentais das principais classes utilizadas pelo protótipo, ilustradas na Figura 13. Estes atributos estão listados no Quadro 23.

Quadro 23 - Atributos fundamentais das principais classes do prototipo

Classe	Atributo	Descrição
Pessoa	Id	Identificador único da pessoa.
	codTel	Código.
	Nome	Nome da pessoa.
	dataNascimento	Data de nascimento da pessoa.
	endereço	Endereço da pessoa.
	Email	E-mail da pessoa.
	Cpf	Cpf da pessoa.
Ingresso	Id	Identificador único do ingresso.
	Aluno	Referência para a pessoa.
	Curso	Referência para o curso do ingresso.
	Status	Status do ingresso (Ativo, concluído...)
MestreAluno	Id	Identificar único de matrícula.
	Turma	Referência para a turma da matrícula.
	periodoLetivo	Referência para o período letivo da matrícula.
MestreDisciplina	Id	Identificador único de matrícula na disciplina.
	mestreAlunoo	Referência para matrícula na turma.
	turmaDisciplina	Referência para turma/disciplina matriculada.
AvaliacaoParcialRegistroNota	Id	Identificador único do registro de nota.
	avaliacaoParcialCadastro	Referência para a avaliação.
	mestreDisciplina	Referência para a disciplina da matrícula do aluno.
	Nota	Nota lançada para a avaliação.

Fonte: elaborado pelo autor.

APÊNDICE D – Roteiro e questionário de avaliação da usabilidade

Este Apêndice apresenta o roteiro com o passo-a-passo para a realização dos testes e o questionário referente à usabilidade do protótipo. O Quadro 24 apresenta o questionário referente ao perfil de usuário. No Quadro 25, apresenta-se as instruções para a realização da avaliação do usuário. Por fim, no Quadro 26 consta o questionário referente ao teste de usabilidade executado pelos usuários.

Quadro 24 - Questionário 01 - Perfil do usuário

Questionário 01 – Perfil do usuário

Por favor, responda com sinceridade todas as questões abaixo.

1. Sexo:

- Masculino
- Feminino

2. Idade:

- Entre 18 e 30 anos
- Entre 30 e 40 anos
- Mais de 40 anos

3. Qual seu grau de conhecimento referente ao software MentorWeb?

- Não conheço
- Baixo
- Médio
- Alto

4. Você já utilizou a digitação de notas do sistema atual do MentorWeb?

- Sim
- Não

Fonte: elaborado pelo autor.

Quadro 25 - Instruções para a avaliação do usuário

Instruções para a avaliação do usuário

As instruções descritas deverão ser seguidas para realizar o teste de usabilidade do protótipo. Após realizar o teste, deverá ser respondido o Questionário 02, onde as respostas serão analisadas para avaliar o protótipo. Você pode utilizar o protótipo por um período de 5 minutos para ambientar-se com o novo conceito de lançamento de notas por comandos de voz. Após este período, inicie o processo de lançamento de notas seguindo os passos abaixo:

1. Acessar o sistema MentorWeb, através do endereço disponibilizado, utilizando o *login*: “prof”, senha: “1”.
2. No menu (canto lateral esquerdo), acessar a opção Diário de classe que fica dentro da opção Operação.
3. Informar o curso: “Ciências Econômicas - Bacharelado (ECN)”, e clicar no botão Pesquisar.
4. Clicar no ícone da coluna Provas e trabalhos.

A partir deste momento, já está acessando a tela de lançamento de notas.

5. Clique no ícone microfone, ao lado do botão Gravar para ativar o comando de voz.
6. Fale: período segunda avaliação. Neste momento, na opção Período, o protótipo deverá selecionar “Segunda avaliação”.
7. Fale: avaliação quinta avaliação. Neste momento, na opção Avaliação, o protótipo deverá selecionar “Quinta avaliação”.
8. Fale: **Aluno** e logo em seguida, o nome do primeiro aluno da lista. Neste momento, o protótipo deverá colocar o foco no campo de texto referente ao primeiro aluno da lista.
9. Fale: **Nota** e logo em seguida, uma nota para o aluno. Neste momento, o protótipo deverá inserir no campo que está com o foco, a nota informada.
10. Repita os passos 8 e 9 com todos os alunos da lista. Ao mencionar o nome do aluno, pode-se mencionar tanto o primeiro nome, bem como o nome completo.
11. Para focar no próximo aluno sem a necessidade de mencionar o nome, basta mencionar a palavra chave Próximo. Para focar no aluno anterior, deve-se falar a palavra chave Anterior.
12. Para limpar um valor informado, deve-se mencionar a palavra chave Limpar.
13. Para realizar a gravação, deve-se mencionar a sentença Ação Gravar.

Observação: Foi disponibilizado um Notebook Sony Vaio, e um Smartphone Samsung Galaxy Note para a realização dos testes. Os testes deverão ser realizados em ambos os dispositivos. No smartphone, a cada ação executada, o reconhecimento é interrompido e iniciado automaticamente logo após realizar a operação. Portanto, atente-se ao aviso sonoro emitido pelo próprio navegador, e só comece a falar após este aviso.

Fonte: elaborado pelo autor.

Quadro 26 - Questionário 02 - Teste de usabilidade

Questionário 02 – Questionário referente ao teste de usabilidade

Por favor, responda com sinceridade todas as questões abaixo.

1. Na sua opinião, o protótipo ficou intuitivo e simples de utilizar?

Computador:

 Sim Não

Smartphone

 Sim Não**2. Os comandos de voz foram reconhecidos com sucesso?**

Computador:

 Sim Não A maioria

Smartphone

 Sim Não A maioria**3. Quando o comando de voz não é reconhecido, ou é inválido, o protótipo está emitindo feedback?** Sim Não**4. Em uma escala de 0 a 10, qual a sua nota para o protótipo?**

Computador: _____

Smartphone: _____

5. Você teve alguma dificuldade na utilização do protótipo? Caso sim, cite qual(is).

Computador: _____

_____.

Smartphone: _____

_____.

Agradeço sua participação.

Fonte: elaborado pelo autor.