



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS - CCEA
CAMPUS VII - GOVERNADOR ANTÔNIO MARIZ
CURSO DE LICENCIATURA EM COMPUTAÇÃO**

TARCIANA ABRANTES PALETOT

**PROPOSIÇÃO DA ARQUITETURA DE UM SOFTWARE EDUCATIVO PARA
PROMOÇÃO DA SAÚDE VOCAL DOS PROFESSORES**

**PATOS
2017**

TARCIANA ABRANTES PALETOT

**PROPOSIÇÃO DA ARQUITETURA DE UM SOFTWARE EDUCATIVO PARA
PROMOÇÃO DA SAÚDE VOCAL DOS PROFESSORES**

Monografia apresentada ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Licenciado em Computação.

Área de Concentração: Engenharia de Software

Orientador: Prof. Me. Pablo Ribeiro Suárez

**PATOS
2017**

É expressamente proibida a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano da dissertação.

P163p Paletot, Tarciana Abrantes
Proposição de uma arquitetura de um software educativo para promoção da saúde vocal [manuscrito] / Tarciana Abrantes Paletot. - 2017.
84 p. : il. color.

Digitado.
Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas, 2017.
"Orientação: Pablo Ribeiro Suarez, CCEA".
"Co-Orientação: Profa. Ma. Carolina Soares Ramos, Departamento de ".

1. Informática. 2. Voz. 3. Software educativo. I. Título.

21. ed. CDD 005

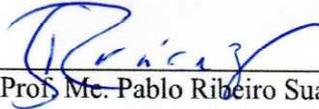
Tarciana Abrantes Paletot

**PROPOSIÇÃO DA ARQUITETURA DE UM SOFTWARE EDUCATIVO PARA
PROMOÇÃO DA SAÚDE VOCAL DOS PROFESSORES**

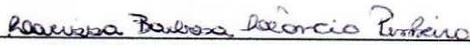
Trabalho de Conclusão de Curso apresentado ao
Curso de Licenciatura em Computação da
Universidade Estadual da Paraíba, em
cumprimento à exigência para obtenção do grau
de Licenciado em Computação

Aprovado em 15 de agosto de 2017

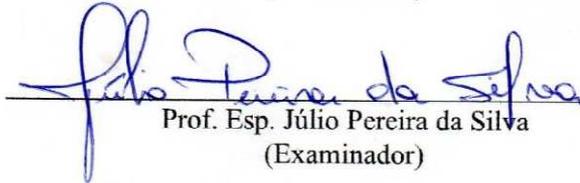
BANCA EXAMINADORA



Prof. Me. Pablo Ribeiro Suárez
(Orientador)



Prof.^a Me. Larissa Barbosa Leôncio Pinheiro
(Examinador)



Prof. Esp. Júlio Pereira da Silva
(Examinador)

Aos meus pais, **Francisca Maria** e **Nazareno** pela minha formação moral, por sua dedicação, incentivo e amor incondicional em todos os momentos. Pela ajuda e apoio em todos os meus projetos de vida. Quero traduzir meu reconhecimento e gratidão, pois sou a continuidade de seus esforços.

Aos meus irmãos, **Germano** e **Yolanda**, pela amizade, colaboração e incentivo.

Ao meu esposo, **Thaciano Victor**, pelo carinho e apoio a mim prestados durante todos os momentos difíceis.

Em especial ao meu filho **Heitor Victor**, que é minha bússola para a direção certa. Por ele quero crescer e ser melhor a cada dia na busca dos meus ideais.

E por fim a **Rafael Borges** (in memoriam) dele tirei sempre o exemplo das maiores qualidades do homem: Dedicação, responsabilidade e honestidade. **Dedico.**

AGRADECIMENTOS

Se você está lendo esse texto é porque eu consegui. E não foi simples estar aqui. Foi uma estrada cheia de obstáculos e nada tranquila. Porém a sola do pé conhece a sujeira da estrada.

Agradeço a Deus por ter me dado saúde e inteligência para superar todas as dificuldades e conseguir chegar onde hoje estou.

Agradeço de forma especial à minha mãe Francisca e meu pai Nazareno, por não medirem esforços para que eu pudesse levar meus estudos adiante, pelo incentivo e apoio incondicional e pela determinação e luta na minha formação.

Aos meus irmãos pela ajuda e incentivo e por mais difícil que fossem as circunstâncias, sempre tiveram paciência e confiança para me ajudar a seguir nessa jornada. E em especial minha irmã Yolanda que idealizou este projeto junto comigo.

Ao professor, orientador e amigo Pablo Ribeiro Suárez por ter acreditado e não ter desistido de mim e que com muita paciência e atenção dedicou seu tempo e ajuda e assim tornou possível a conclusão desse trabalho.

À minha co-orientadora Carolina Ramos pela dedicação e ajuda, que foram de suma importância para o sucesso desse trabalho.

Ao meu esposo Thaciano, que de forma especial e carinhosa me deu força e coragem, me apoiando nos momentos de dificuldades.

Ao meu filho Heitor que embora não tenha conhecimento disto, me ilumina de uma maneira muito especial e me leva a buscar mais conhecimentos.

A todos os meus amigos que, mesmo não estando citados aqui, tanto contribuíram para a conclusão dessa fase da minha vida.

A Rafael, meu anjo na terra (in memoriam), foi alguém que me ensinou com o exemplo que é necessário dar sempre o melhor em tudo que me propuser a realizar. Me perdoe se não cumpri os nossos sonhos e falhei no caminho. Mas Deus é muito bom e te trouxe para a minha vida e você vive em mim e viverá sempre.

A vocês, muito obrigada!

“Os sonhos vêm e os sonhos vão o resto é
imperfeito... Meu amor, disciplina é
liberdade, compaixão é fortaleza ter bondade é
ter coragem.”

Renato Russo

RESUMO

A voz do professor é apontada por ele mesmo como um de seus principais recursos de trabalho, porém pela falta de treinamento vocal, e por um conjunto de condições desfavoráveis de ensino, o professor torna-se um profissional de risco para desenvolver algum problema de voz. A informática tem servido como instrumento de armazenamento, processamento e recuperação da informação em todas as áreas do conhecimento. Este trabalho tem como objetivo principal o desenvolvimento de uma proposição de software para promoção da saúde vocal de professores, com a proposta de difundir os conhecimentos acerca de saúde vocal. Esta proposição foi desenvolvida a partir de duas fases: a revisão de literatura que dá embasamento a pesquisa e a construção da proposição a partir da metodologia YPeduc, o que possibilita maior sucesso nesse tipo de implementação pela conotação educacional. A proposição da arquitetura, mesmo que em um estágio embrionário, induziu a construção de protótipos que atenderam às expectativas da modelagem dos requisitos, gerando, de certo modo, resultados satisfatórios e permitindo que o aplicativo naturalmente entre em produção, devendo ser implementados e aprimorados no futuro. As contribuições apresentadas já representam um certo grau de avanço dentre os mais variados usos de software educacional, neste caso específico, voltado para a saúde vocal.

Palavras-Chave: Informática. Voz. Docentes. Programas.

ABSTRACT

The teacher's voice is pointed to by himself as one of its key features of work, however the lack of vocal training, and a set of unfavourable conditions of teaching, the professor becomes a risk professional to develop vocal disorders. It has served as a tool for storage, processing and retrieval of information in all areas of knowledge. This work aims to spread the knowledge of vocal health. The proposition of architecture, even if in an embryonic stage, led the construction of prototypes that answer the expectations of the modeling requirements, generating, in a sense, satisfactory results and allowing the application of course between in production and should be implemented and enhanced in the future. The contributions represent a certain degree of progress among the various uses of educational software, in this specific case, focused on vocal health.

Keywords: Computing, Voice, Teachers, Software.

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 – Camadas da Engenharia do Software | 23 |
| Figura 2 – Fluxo do Processo YPEduc | 28 |
| Figura 3 – Processo e Informação em Medicina | 32 |
| Figura 4 – Modelagem da Tela 1 | 43 |
| Figura 5 – Modelagem da Tela 2 | 43 |
| Figura 6 – Modelagem da Tela 3 | 44 |
| Figura 7 – Modelagem da Tela 4 | 44 |
| Figura 8 – Modelagem da Tela 5 | 45 |
| Figura 9 – Modelagem da Tela 6 | 45 |
| Figura 10 – Modelagem da Tela 7 | 46 |
| Figura 11 – Modelagem da Tela 8 | 46 |
| Figura 12 – Tela Inicial | 52 |
| Figura 13 – Tela 2 Cuidados com a Voz | 52 |
| Figura 14 – Tela 3 Inimigos da Voz | 53 |
| Figura 15 – Tela 4 Cuidados com a Voz | 53 |
| Figura 16 – Tela 5 Voz Saudável, Comunicação em dia | 54 |
| Figura 17 – Tela 6 Voz Profissional | 54 |
| Figura 18 – Tela 7 Buscar Profissionais | 55 |
| Figura 19 – Perguntas a Fazer ao Elaborar um Projeto Arquitetural | 57 |
| Figura 20 – Modelo Lógico de Dados | 58 |

LISTA DE QUADROS

| | |
|---|----|
| Quadro 1 Princípios do Manifesto Àgil..... | 26 |
| Quadro 2 Identificação do Escopo do Problema | 36 |
| Quadro 3 Definição de Papéis | 36 |
| Quadro 4 Descrição do Sistema | 38 |
| Quadro 5 Definição dos Requisitos | 39 |
| Quadro 6 Perfil do Professor | 4 |
| Quadro 7 Formulário para o levantamento do Perfil Profissional da Saúde | 41 |
| Quadro 8 Objetivos de Usabilidade do Yvoice | 42 |
| Quadro 9 <i>UserStories</i> e Testes de Aceitação | 47 |
| Quadro 10 Projeto Arquitetural | 56 |
| Quadro 11 Projeto Arquitetural (continuação) | 56 |
| Quadro 12 Perguntas a Fazer ao Elaborar um Projeto de Arquitetura | 57 |
| Quadro 13 Plano de <i>Realse</i> | 59 |
| Quadro 14 Plano de Interação | 60 |
| Quadro 15 Implementação do Projeto | 62 |
| Quadro 16 Teste de Usabilidade | 64 |
| Quadro 17 Desenvolvimento do API do Sistema | 74 |
| Quadro 18 Implementação, Testes e Validação das Funcionalidades: Informações sobre Anatomia do Aparelho Vocal | 75 |
| Quadro 19 Implementação, Testes e Validação das Funcionalidades: informações sobre os inimigos da voz na API | 76 |
| Quadro 20 Implementação, Testes e Validação das Funcionalidades: cuidados com a voz na API | 77 |
| Quadro 21 Implementação, Testes e Validação das Funcionalidades: voz saudável, comunicação em dia na API | 77 |
| Quadro 22 Implementação, Testes e Validação das Funcionalidades: voz profissional na API | 79 |
| Quadro 23 Implementação, Testes e Validação das funcionalidades: a funcionalidade manter profissionais na API | 80 |
| Quadro 24: Implementação, Testes e Validação das Funcionalidades: guia simples de ajuda | 80 |
| Quadro 25: Prototipação das Telas Desenvolvidas Estaticamente no IONIC | 81 |
| Quadro 26: Integração entre <i>Frontend</i> e <i>Backend</i> da Funcionalidade: informações sobre anatomia do aparelho vocal | 82 |
| Quadro 27: Integração entre <i>Frontend</i> e <i>Backend</i> da Funcionalidade: informações sobre os inimigos da voz | 83 |
| Quadro 28: Integração <i>Frontend</i> e <i>Backend</i> da Funcionalidade: informações sobre os cuidados com a voz | 83 |
| Quadro 29: Integração entre <i>Frontend</i> e <i>Backend</i> da Funcionalidade: informações sobre voz saudável, comunicação em dia | 84 |
| Quadro 30: Integração entre <i>Frontend</i> e <i>Backend</i> da Funcionalidade: informações sobre voz profissional | 85 |
| Quadro 31: Integração entre <i>Frontend</i> e <i>Backend</i> da Funcionalidade: manter profissionais | 86 |
| Quadro 32: Integração entre <i>Frontend</i> e <i>Backend</i> da Funcionalidade: guia simples de ajuda..... | 87 |
| Quadro 33: Validação das Integrações Realizadas | 87 |

SUMÁRIO

| | |
|--|-----------|
| 1 INTRODUÇÃO | 11 |
| 1.1 CENÁRIO TÉCNICO-CIENTÍFICO | 11 |
| 1.2 DEFINIÇÃO DA PROBLEMÁTICA | 12 |
| 1.3 OBJETIVOS | 14 |
| 1.3.1 Objetivo Geral | 14 |
| 1.3.2 Objetivos Específicos | 14 |
| 1.4 JUSTIFICATIVA..... | 15 |
| 1.5 METODOLOGIA | 16 |
| 1.6 ESTRUTURA DO TRABALHO..... | 17 |
| 2 DESENVOLVIMENTO | 18 |
| 2.1 ENGENHARIA DE SOFTWARE EDUCACIONAL PARA A SAÚDE | 18 |
| 2.2 SOFTWARES EDUCACIONAIS | 18 |
| 2.2.1 Tipos de <i>Software</i> Educacional | 19 |
| 2.2.2 Níveis de Aprendizagem | 21 |
| 2.2.3 Aspectos Técnicos | 21 |
| 2.3 ENGENHARIA E DESENVOLVIMENTO DE SOFTWARE..... | 22 |
| 2.3.1 Engenharia de Software Educacional | 24 |
| 2.4 METODOLOGIAS ÁGEIS | 26 |
| 2.4.1 Ypeduc | 27 |
| 2.5 INFORMÁTICA EM SAÚDE..... | 32 |
| 2.5.1 Promoção da Saúde Vocal do Professor | 34 |
| 2.6 DISFONIA | 34 |
| 2.7 ALTERAÇÕES VOCAIS NOS DOCENTES | 35 |
| 3. MINIPROJETO PARA O DESENVOLVIMENTO DO SOFTWARE EDUCATIVO: YVOICE | 36 |
| 4 CONCLUSÕES E CONSIDERAÇÕES FINAIS | 67 |
| 4.1 CONTRIBUIÇÕES | 67 |
| 4.2 LIMITAÇÕES..... | 68 |
| 4.3 TRABALHOS FUTUROS..... | 69 |
| REFERÊNCIAS | 70 |
| ANEXO | 74 |

1 INTRODUÇÃO

O presente tópico tem por intuito apresentar o cenário onde se desenvolve o tema, a problemática de pesquisa, objetivos que norteiam o trabalho, bem como as justificativas para a realização do estudo.

1.1 CENÁRIO TÉCNICO-CIENTÍFICO

A tecnologia tem influenciado continuamente o cenário educacional ao passo que a prática educativa se faz cada vez mais presente aumentando a necessidade de recursos de qualidade na área. Esta é uma época de desafios tecnológicos que permitem mais desenvolvimento e acesso a informação.

Pode-se afirmar que as tecnologias digitais da informação e comunicação (TDIC) foram inseridas no processo de ensino aprendizagem e que deste nunca serão desvinculadas. Assim, os recursos tecnológicos têm influência direta na carreira de muitos profissionais, não sendo diferente com a figura do professor. Porém, é preciso adaptar-se a esse mundo novo e aprimorar os conhecimentos, bem como a correta utilização pedagógica dessas diversas ferramentas tecnológicas.

São inúmeras as ferramentas que podem dar auxílio em relação à aprendizagem, daí pode-se incluir os *softwares* educacionais, que servem como grandes aliados, assim como afirma Barreto (1999), eles auxiliam no processo de desenvolvimento da capacidade de aprender a aprender e personalizando a construção de conhecimentos no processo de aprendizado contínuo.

Os empreendimentos na área de softwares educativos crescem a cada dia e causam um grande impacto na maneira como se transmite a informação e como se relacionam essas informações a fatos e conhecimentos. O professor sai do papel de mediador e também assume a condição de aprendiz diante da indústria do *software*. Entretanto, é preciso analisar que é fundamental que o *software* educativo consiga construir uma concepção teórica em cima de uma aprendizagem que oriente para transformar o usuário em um sujeito que aprende e constrói seu próprio conhecimento.

Para Sommerville (2007) o processo de desenvolvimento de uma aplicação de software tem início no levantamento de requisitos, incidindo, assim, em uma série de etapas fundamentais para concepção de uma aplicação de qualidade. Wazlawick (2013) ainda complementa dizendo que a documentação das etapas auxilia na

sincronização da equipe de trabalho, norteando e organizando o processo de *software*. Portanto, os recursos educacionais devem ser apropriados para se alcançar à qualidade do *software*, estando este em harmonia com requisitos funcionais e de desempenho interligados pedagogicamente as metas estabelecidas e as dificuldades encontradas.

Diante disso observa-se que os estudos envolvendo processos de desenvolvimento de software, mais especificamente na área da saúde, vem ganhando muito espaço. É sabido que o arranjo entre educação, tecnologia e saúde beneficia diretamente a qualidade do processo de vivência dos usuários influenciando os métodos, as práticas e os recursos dessas áreas. Dessa forma, utilizá-las de uma maneira mais apropriada traz grandes benefícios como um ganho significativo na assimilação e interpretação de informações, bem como induz a resolução de problemas em tempo ágil, dando ao usuário a possibilidade de melhorar a partir dos erros e não só dos ganhos.

Assim, partindo desse pressuposto, entende-se que a correta ligação entre os recursos tecnológicos e a saúde humana traz um bem-estar psíquico, físico e social, estabelecendo conexões e abrindo portas para as mais diversas aplicações. Portanto, a junção de *software* educacional e saúde, mais especificamente tendo como área de concentração deste estudo aspectos relacionados a fonia (a voz), se concretizam na proposição de uma arquitetura que consiga unir de forma substancial os benefícios das metodologias ágeis, sistematizando assim um *software* educativo nessa área, quebrando o paradigma das abordagens tradicionais.

1.2 DEFINIÇÃO DA PROBLEMÁTICA

A educação transforma a consciência e centraliza o indivíduo no mundo, de uma forma crítica e autônoma devendo, assim, adquirir a capacidade de se adequar as mudanças e aos ambientes informatizados.

Todo o processo tecnológico auxilia na aquisição de conhecimento e torna tanto o ensino quanto a aprendizagem mais estimulante, fazendo com que o usuário adicione ao seu perfil novas exigências, cada vez mais complexas. Observa-se, então, que os instrumentos do processo tecnológico podem ter participação significativa nessa contínua transformação tanto no aspecto pessoal quanto no aspecto profissional.

O maior instrumento de trabalho do professor, bem como em diversas outras profissões, é a voz. Muito se tem debatido sobre o assunto no tocante ao uso de tecnologias para sua aplicação prática em diversas áreas de pesquisa, de modo a

contribuir para o conhecimento e para o estudo, detalhadamente, da anatomia e da fisiologia da voz humana.

A fonoaudiologia objetiva estudar a comunicação humana em suas manifestações normais e patológicas sendo a voz uma ferramenta importante para esta comunicação, sendo analisada em diversos profissionais, dentre eles, o professor.

Segundo Behlau e Zambon, (2009), um problema de voz pode vir a interferir na própria habilidade de se comunicar necessitando de uma abordagem fonoaudiológica correta para a recuperação parcial ou total da voz, uma vez que estas alterações incidem sobre a vida profissional já que a voz é o “cartão postal” do professor, assumindo esta um papel fundamental tanto no âmbito social quanto no psicológico.

Ainda há docentes que tem pouca noção da influência da voz no desempenho de sua função, sendo a mesma o seu principal recurso de comunicação e de transmissão de conhecimentos. A falta de conhecimento, mesmo o básico, sobre os cuidados com este mecanismo leva esses profissionais a diversos distúrbios vocais. Quando a voz começa a apresentar problemas, falhar ou apresentar sinais de fadiga é que estes docentes tomam consciência da importância da própria voz. Porém, em muitos casos, já está estabelecida uma ou mais patologias que o impede de desenvolver suas atividades acadêmicas.

Um fato a ser levado em consideração é que o professor é um formador de opiniões, desempenhando um importante papel na educação de seus alunos, portanto, os cuidados com a voz e todas as repercussões negativas dos problemas atrelados a esta, são motivos suficientes para a realização deste estudo. A tecnologia pode ajudar no sentido de educar e melhorar a qualidade de vida de diversos profissionais. Por sua vez, os estudos de *softwares* na área da saúde têm crescido exponencialmente, diante da necessidade de adequar a tecnologia a uma melhora na qualidade de vida.

A construção de *softwares* voltados para a área da saúde tem sido uma prática bastante difundida, estando fortemente conectada às duas áreas e contribuindo para a melhoria de equipamentos, métodos e procedimentos, obtendo, assim, uma maior qualidade no que se refere aos cuidados com a saúde. Segundo Pinochet (2011), a era da informação não deixou a área da saúde na mão e, diante desses grandes avanços tecnológicos, pode-se influir que tanto as pessoas quanto as organizações da área da saúde terão como benefício uma grande evolução e transformação.

Para a concretização de projetos no cenário atual de desenvolvimento de *software*, é indispensável citar o manifesto ágil de *software*, que se contrapõe às

metodologias clássicas de desenvolvimento. O confronto se destaca em dois conceitos, uma vez que o movimento ágil pensa nas pessoas e em interações, na construção de *software* executável com a colaboração do cliente e respostas rápidas para a mudança. Já a metodologia tradicional traz processos e ferramentas com documentações muito extensas e longas negociações tendo que seguir roteiros previamente definidos.

Dentre as metodologias ágeis existentes destaca-se o YP – *easYProcess*(PROGRAMA DE EDUCAÇÃO TUTORIAL, 2007), que é uma metodologia criada por meio de um projeto da Universidade Federal de Campina Grande (UFCG). Essa estrutura visa a produção de projetos acadêmicos, de forma mais leve e com melhor assimilação do aprendizado. O YP também pode ser adaptado a diversos cenários, dentre os quais destaca-se o educacional. Portanto, o ajuste que melhor se enquadra nessa pesquisa foi realizado por Medeiros (2012) na construção do Ypeduc, processo que adapta as etapas tradicionais ao contexto de *software* educacional.

Em contraposição ao pensamento em termos da tecnologia ou do programa (ou aplicativo) em si, deve-se antes compreender que tipo de dificuldade são enfrentadas por pessoas que tem problemas vocais e pensar em como a tecnologia pode contribuir para a resolução desses problemas. Pode-se focar, então, o seu uso para a educação ou mais especificamente para a instrução a ser proporcionada aos usuários. Isso, conseqüentemente, leva a seguinte reflexão: É possível a proposição de uma arquitetura para um sistema (programa ou aplicativo) que promova a saúde vocal de professores utilizando o Ypeduc como processo de desenvolvimento? Para se ter alguns elementos de resposta tão desejados para esta problemática levantada são necessários serem traçados os objetivos dessa pesquisa, tanto em caráter geral como em caráter específico.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Este trabalho tem como objetivo principal elaborar uma proposição arquitetural de *software* educativo especializado em cuidados vocais, afim do mesmo tornar-se uma fonte instrucional para profissionais do uso da voz, com prioridade aos professores.

1.3.2 Objetivos Específicos

- Estudar a literatura acerca do tema escolhido;
- Adotar o processo de desenvolvimento YPEduc para o planejamento e para a execução das tarefas que circunscrevem o desenvolvimento de um software educativo;
- Propor uma arquitetura para um *software* educativo especializado em saúde vocal, com base no roteiro preconizado pelo YPEduc.

1.4 JUSTIFICATIVA

Não só computadores, mas *tablets*, *smartphones* e diversos aparelhos tecnológicos se tornaram um recurso poderoso para acesso à informação. Sua adequação às mais diversas áreas favorece o aprendizado, uma vez que tais recursos tecnológicos se aplicam em diversas situações. Nesse contexto o professor está adquirindo uma nova face e valor no mercado das profissões, já que o mesmo assume um papel diversificado dos demais, pois diante da tecnologia assume tanto o papel de educador quanto de aprendiz, tendo que estar sempre atualizado no manuseio de tais aparelhos.

A relação do professor com o seu trabalho vem sendo ajustada desde os primórdios. A preocupação com sua saúde e com seu bem-estar geral muitas vezes ficou em segundo plano. No entanto, o contexto globalizado em que este profissional se encontra, com flexibilização nos processos de trabalho e um apurado tratamento e valorização de informações, trouxe maior produtividade e maior qualidade nesse processo, uma vez que o rendimento do profissional depende diretamente de sua saúde e bem-estar.

Dentre os profissionais da voz, o professor é um dos mais susceptíveis a ter um distúrbio vocal em decorrência da multifatorialidade existente no ambiente de trabalho. Segundo Behlau et. al. (2004, p. 32), "a voz é um componente importante na comunicação interpessoal, uma vez que transmite palavras, mensagens e sentimentos, devido a isso, torna-se responsável pelo sucesso das interações humanas, tanto em âmbito privado quanto em âmbito profissional". O distúrbio de voz tem levado vários profissionais da voz a situações de afastamento e incapacidade para o desempenho de suas atividades, o que implica custos financeiros e sociais.

Observa-se, então, que a utilização de *softwares* educacionais nessa área pode auxiliar de forma substancial a qualidade de vida profissional do docente. A escolha por

essa pesquisa deu-se pela possibilidade de elaborar uma ferramenta tecnológica para instruir o uso correto da voz e alertar para os possíveis problemas na falta de cuidado com a mesma. Portanto, estas orientações para a preservação do maior instrumento de trabalho daquele que está acostumado a ensinar funciona como artefato de motivação. A partir dessa explanação, é de extrema importância a criação de uma proposta que olhe para este profissional e o oriente corretamente para a melhor abordagem fonoaudiológica para a voz.

1.5 METODOLOGIA

Sabe-se que através da pesquisa pode-se obter uma apuração de problemas teóricos ou práticos usando corretamente os processos científicos. Na manipulação desses recursos destaca-se que a pesquisa não se trata apenas de ampliação de competências, mas da incorporação de novas relações que difunda o exercício compartilhado para tornar produtivas as interações entre os profissionais e o conhecimento gerado.

A criação de aplicativos na área da saúde oferece suporte não só para os profissionais, mas também para a população em geral contribuindo para a educação em saúde, na escolha da terapia mais apropriado, no diagnóstico e no tratamento de enfermidades, bem como na prevenção de problemas futuros.

Segundo Pinochet (2011), a era tecnológica chegou abrangendo diversas áreas entre elas muitos setores da saúde e as instituições seguiram as novas técnicas operacionais onde foram deixados de lado tantos processos obsoletos e a tecnologia da informação acabou trazendo inúmeras facilidades e melhoria nos diversos segmentos.

Pode-se destacar que o celular tem múltiplas funções e pode ser visto como um computador de bolso e, conseqüentemente, leva o usuário a ter acesso a milhões de aplicativos, fabricados em todo o mundo e isso traz uma maior capacidade de mudança e quebra de várias limitações. O *smartphone* faz parte da vida do usuário e pode acompanhá-lo 24 horas por dia em qualquer lugar, tendo uma poderosa arma para a assistência na saúde e no acesso a informação.

Marcon e Dias (2014) afirma que, atualmente, as pessoas passam muito mais tempo ao celular e conectadas à internet, com acesso a informações em tempo real, interagindo não apenas em suas residências, mas em qualquer lugar e a qualquer hora. É exponencial o crescimento do número de pessoas que passam cada vez mais tempo

interligadas pela rede, o que contribui diretamente para o uso e a venda tanto de aparelhos quanto de aplicativos, uma vez que estes invadiram as mais diversas áreas, implicando em cada vez mais entusiastas no ciberespaço.

Dessa forma a presente pesquisa exposta destina-se a aquisição de conhecimento em uma condição ímpar, além disto, como um dos objetivos deste trabalho é construir a proposição de uma arquitetura que seja capaz de auxiliar o professor quanto a saúde vocal utilizando o YPEduc, é de extrema importância o estudo exploratório, já que o mesmo possibilita maior familiaridade com o problema apresentado.

1.6 ESTRUTURA DO TRABALHO

Primeiramente o trabalho apresenta a Introdução, trazendo o tema da pesquisa e sua abordagem relacionada à concepção de uma proposição de arquitetura de *software* voltada para os cuidados com a voz dos professores, bem como o cenário técnico-científico, a problemática, os objetivos, a justificativa, e a metodologia empregada e, ainda, a organização estrutural do trabalho.

No capítulo 2 encontra-se o estado de conhecimento, que será construído a partir dos estudos bibliográficos já realizados, dando ênfase a reflexão e a teorização para a construção da proposição.

Por sua vez, no capítulo 3 será apresentado um estudo de caso, com o uso do YPEduc em um projeto de desenvolvimento real, para a proposição de uma arquitetura de um *software* educacional voltado para a saúde vocal dos professores, na forma de um aplicativo. Por fim, no capítulo 4 encontram-se as conclusões obtidas com a realização do trabalho, suas contribuições e limitações, bem como as sugestões para trabalhos futuros.

2 DESENVOLVIMENTO

2.1 ENGENHARIA DE SOFTWARE EDUCACIONAL PARA A SAÚDE

Esta seção está dividida em quatro subseções. Na primeira, são apresentados conceitos de *software* educacional, seus tipos, níveis de aprendizagem e aspectos técnicos. Na segunda subseção, o tema abordado é engenharia e desenvolvimento de *software* e engenharia de *software* educacional, ressaltando a importância de utilizar técnicas e métodos específicos em projetos de desenvolvimento de *software* educativo. Na terceira, são apresentados os conceitos que envolvem as metodologias ágeis e o YPeduc. Por fim na quarta subseção estão os conceitos de informática em saúde, promoção da saúde vocal do professor e os *softwares* na área de voz.

2.2 SOFTWARES EDUCACIONAIS

A computação se instalou de modo irreversível na nossa vida. Portanto, é essencial refletir sobre seu aproveitamento na educação. Partindo dessa realidade, existem aspectos que são indispensáveis para a utilização da informática na educação: o uso do computador ou dispositivo móvel, o tipo de *software*, como o professor recebe e utiliza esse programa e como o aluno ou aprendiz é transformado por essa tecnologia.

Silva (2016) afirma que a sociedade atual é marcada pelos avanços tecnológicos que desenvolveram muito rápido a comunicação e a informação. Como exemplo ele diz que o conhecimento que se tornou de fácil acesso em todas as áreas para todos. Porém, observa-se que muitas pessoas não estão preparadas para essa realidade social e histórica promovida pela revolução tecnológica.

Faz-se notar que a perspectiva da informática educativa não é apenas o uso do computador como único recurso de aprendizagem, hoje os dispositivos móveis como *tablet* e *smartphones* podem ser vistos como forma de aquisição de conhecimentos, e dependendo da utilização são grandes aliados na obtenção de conhecimento, pois o aprendizado através dessa gama de dispositivos mostra que o

usuário tem um caminho multifacetado e adquire informação sobre diversas áreas servindo-se dos recursos tecnológicos.

A utilização do *software* na educação ganha força ao passo que os professores enxergam uma ferramenta que seja capaz de auxiliar as suas atividades didático-pedagógicas, além de ser um poderoso instrumento de auxílio para projetos e planejamento, pois motiva e desafia as novas práticas pedagógicas.

Mugnaini (2008) ainda complementa dizendo que o *software* educacional é um programa que pode ser utilizado na administração escolar ou em contextos pedagógicos, ou como recurso de aprendizagem e é caracterizado pela sua admissão em contextos educacionais, e o *software* educativo é uma classe incluída dentro deste, que possibilita aprendizagem ativa que conduz o usuário a construir conhecimento, este é o programa construído para a educação.

A intenção de um programa educativo é a de beneficiar as técnicas de ensino aprendizagem e seu atributo principal é seu caráter didático. A seleção do *software*, além de ser de fundamental importância para o trabalho que o professor irá desenvolver com seus alunos, implica uma visão de mundo, uma compreensão de educação. Então, assim podemos destacar a importância que deve ser dada à escolha dos *softwares* para serem usados com a finalidade educacional.

A sociedade do conhecimento requer indivíduos criativos e com a capacidade para criticar construtivamente, pensar, aprender sobre aprender, trabalhar em grupo e conhecer seus próprios potenciais. Esse indivíduo precisará ter uma visão geral sobre os diferentes problemas ecológicos e sociais que preocupam a sociedade de hoje, bem como profundo conhecimento em domínios específicos. Isso requer um indivíduo que está atento às mudanças que acontecem em nossa sociedade e que tem a capacidade de constantemente melhorar e depurar suas ideias e ações. (VALENTE, 1999b, p. 83).

Vários estudos têm demonstrado que as tecnologias na educação, especificamente *softwares* educativos, empregados como um meio didático, ou instrumento de ensino, colaboram de maneira expressiva para métodos educacionais em qualquer grau de ensino. A educação precisa buscar novos meios de adaptação às necessidades de uma sociedade transformada e tecnológica.

2.2.1 Tipos de *Software* Educacional

Entre os diferentes tipos de *softwares* empregados na educação os *softwares* educacionais podem ser qualificados em algumas categorias, de acordo com a sua

utilidade pedagógica. Segundo Valente (1999) são eles: Tutoriais, exercícios e práticas, programação, aplicativos multimídia, internet, simulação e modelagem, jogos e tutores inteligentes (sistemas especialistas), sendo detalhados a seguir:

A cerca dos **tutoriais**, eles têm por característica principal conduzir informações pedagogicamente organizadas, a informação é apresentada ao aluno seguindo um processo, e ele pode escolher a informação que desejar. A mensagem que está disponível para o aluno é definida e formada antecipadamente, assim o computador toma o papel de um instrumento de ensinar.

Já os **exercícios e práticas**, ressaltam a exposição das lições ou exercícios como o próprio nome indica, e nesse caso, o *software* segue um padrão semelhante ao de muitos livros didáticos. Uma informação é repassada ao estudante e um *software* faz questionamentos sobre o determinado conhecimento.

Analisando os **softwares de programação** podemos afirmar que permitem que professores ou alunos criem seus próprios protótipos de programas, pois existe um processamento da informação e ela é transformada em conhecimento, e para isso as pessoas que utilizam este tipo de *software* não precisam ter conhecimentos avançados de programação.

Já os **aplicativos**, são programas voltados para aplicações representativas, como processadores de texto, planilhas eletrônicas, e gerenciadores de banco de dados. Inicialmente foram criados para utilização na educação, podendo se adaptar a diversas áreas.

No que se refere a **multimídia e Internet**, a multimídia e hipermídia utilizam diferentes mídias para representar as informações, podendo incluir fotos, sons, vídeos, animação etc. e são comumente definidos como forma não linear de armazenamento de informações e disponibilizam uma grande quantidade de material de aprendizado estruturado. Podemos citar a internet, uma rede mundial de recursos tecnológicos que conecta computadores de todos os tipos e capacidades utilizando protocolos e proporcionando ao usuário uma enorme quantidade de informação e possibilidades de acesso a serviços diversificados por meio de páginas ou sites.

Simulação e Modelagem é um grande atrativo do uso do computador na escola, pois permitem a representação das mais diversas situações serem refletidas em aula. A simulação acontece com a implementação do objeto desejado no computador, isso se torna ferramenta poderosa para assimilação de conteúdos e é um objeto lúdico na sala de aula.

Jogos trazem uma interface de desafio ou competição, com várias propriedades motivacionais que podem ser individuais ou em grupo. Os jogos são um grande desafio para a educação, pois despertam a curiosidade e a fantasia, porém não podem perder o seu uso educacional, e também levam o aluno não só a competições saudáveis, mas também afluem a cooperação com os colegas e o reconhecimento de suas atividades.

Os tutores inteligentes ou sistemas especialistas levam em consideração o entendimento e aptidões do usuário para traçar metas de ensino que mais se adaptam ao perfil de cada um. Tutores inteligentes consideram o embasamento de conhecimento do especialista no assunto em questão, neste tipo de *software*, a interface deve ser apropriada para oferecer diferentes tipos de ambiente de aprendizagem, ser flexível, e permitindo ao usuário encontrar com facilidade o conhecimento pretendido. Então com essas propriedades, o sistema deve ter um módulo tutor, que cria ambientes para solução de situações problemas.

2.2.2 Níveis de Aprendizagem

Usando os conceitos de níveis de aprendizagem os *softwares* são classificados em três níveis de aprendizagem segundo Vieira (1999) são: nível sequencial, relacional e criativo. O *software* é sequencial quando a preocupação é se o usuário memoriza o conteúdo apresentado e o repete quando solicitado, então este nível de instrução leva o aluno a ser um sujeito passivo. Já no nível relacional o intuito é a obtenção de conhecimento e habilidades e existe uma ênfase maior com diversos acontecimentos e com várias fontes de informação; acontece assim uma influência direta e a interação é homem/máquina, só que este nível acaba levando o aluno ao isolamento. Podemos dizer que o programa é criativo quando ele interliga novos esquemas mentais e gera a interação entre pessoas e tecnologias, contribuindo para um aprendizado participativo.

2.2.3 Aspectos Técnicos

Outro fator importante de avaliação do *software* educacional segundo Rocha (2001) são os aspectos técnicos, pois estes aspectos auxiliam o usuário para uma adequada utilização. Então podemos citar os seguintes aspectos:

- Características pedagógicas que mostram a possibilidade do uso em situações educacionais;

- As características de interface: as mídias que serão empregadas, a qualidade de telas, analisar se o *software* tem uma interface amigável e disponível;
- A sua adaptabilidade: adequando-se as diversas situações dando opção para o usuário e se encaixando nas diversas situações educacionais;
- Documentação: a facilidade de uso e de instalação e desinstalação e manipulação do *software* e averiguar os recursos hipertexto e hiperlink, considerar se o manual técnico tem uma linguagem apropriada ao professor ou usuário;
- Sua portabilidade: Capacidade de ser instalado em diferentes equipamentos, ver a possibilidade de compartilhamento em rede local e *Internet*, também se ele é compatível com outros *softwares*, *hardware* e funcionalidade na rede;
- E por fim a análise do retorno do investimento realizado que demonstra sua adequação ao mercado.

2.3 ENGENHARIA E DESENVOLVIMENTO DE SOFTWARE

O *software* é uma peça fundamental para a realidade tecnológica que nos encontramos, pois entre outras coisas ele tem o poder de manipular a informação. Pode-se concluir que o *software* é o conjunto de vários artefatos e não apenas o código fonte (SOMMERVILLE, 2003). Os seguimentos computacionais estão em constante mudança e dentre eles podemos destacar o crescimento da engenharia de *software*, que nada mais é do que um tratamento ordenado e disciplinado para o desenvolvimento de *software*.

A partir daí podemos começar a entender que a engenharia de *software* não foca apenas no desenvolvimento, mas também em padrões de qualidade, com prazos adequados e custos acessíveis. É uma ramificação da engenharia que se relaciona com os mais diferentes aspectos de elaboração de *software*, desde a especificação até a manutenção. A engenharia de *software* ainda está diretamente ligada com procedimentos de gerenciamento de projeto, desenvolvimento de ferramentas, métodos e teorias que dão suporte a produção.

Para tanto, engenharia de *software* é decorrente da engenharia de sistemas e de hardware, porém podemos diferenciar a engenharia de *software* da engenharia de sistemas dizendo que esta última se preocupa com os aspectos do desenvolvimento e da evolução de sistemas complexos, ela está ligada ao desenvolvimento de *hardware*,

políticas e processos de implantação, já a engenharia de *software* está envolvida com o conceito de arquitetura e integração de partes.

Pressman (2005, p. 24) diz que um processo de desenvolvimento de *software* pode ser considerado como sendo “uma coleção de padrões que definem um conjunto de atividades, ações tarefas de trabalho, produtos e/ou comportamento relacionados necessários ao desenvolvimento de *softwares* de computador”

Para Pressman (2005), a Engenharia de *Software* pode ser dividida em três camadas: processos, métodos e ferramentas. E a base de todas essas camadas é a qualidade do *software* desenvolvido.



Figura 1: Camadas da engenharia do *Software*

Fonte: Elaborada pelo autor adaptado de Pressman (2005).

1. Processos: É o elo que mantém unidos os métodos e as ferramentas e permite o desenvolvimento lógico e mais adequado do *software*. Os processos são à base da engenharia de *software*.

2. Métodos: Um método é uma abordagem estruturada e traça um caminho para a construção de *software*, objetivando assim uma produção de *software* de qualidade elevada com custos acessíveis. Não podemos dizer que existe um método ideal, e sabe-se que diferentes métodos possuem diferentes áreas onde são mais aproveitáveis, e os métodos são baseados na ideia de modelos de desenvolvimento.

3. Ferramentas: Elas dão suporte ao desenvolvimento do *software* e dão apoio automatizado ou semi-automatizado aos métodos, as ferramentas CASE (*Computer-Aided Software Engineering*). São utilizadas para dar apoio às atividades do processo de *software*, e dão apoio à análise, modelagem, depuração e testes. Todos os métodos vêm atualmente com uma tecnologia case associada e as ferramentas combinam aplicações em *software, hardware* e um banco de dados.

No tocante a processo de *software*, segundo Sommerville (2007) são as fases necessárias para resolver os problemas do mesmo, ele diz ainda que o processo é um conjunto de atividades e resultados associados que produz um produto de *software*. E as atividades pertinentes aos processos são a especificação que é nada mais do que a definição do que será produzido, o desenvolvimento que é onde o artefato é projetado e programado, a validação onde se verifica o que foi feito e se todas as vontades dos clientes foram atendidas e a evolução que é a adaptação para outras versões ou melhorias.

Sommerville (2007) ainda define modelos de processo de *software* dizendo que é uma apresentação mais compreensível do processo, e neles estão incluídas atividades do processo, os produtos de *software* e por fim os papéis de todos os envolvidos. Observa-se que a grande parte dos modelos se baseia em alguns paradigmas gerais de desenvolvimento, que são eles:

- Cascata – Fases separadas de processos. Uma fase não inicia sem que outra termine e seja aprovada.
- Desenvolvimento iterativo – Intercala as atividades. Desenvolvido rapidamente com base em especificações abstratas e depois é refinado com informações dos clientes para que possa satisfazer a necessidade desses.
- Engenharia de *software* baseada em componentes – Supões que partes do sistema da já existem. Concentra-se mais na integração dessas partes do que no seu desenvolvimento a partir do início.

Os atributos de um *software* de qualidade espelham o comportamento do mesmo, na sua execução, estrutura e documentação. Porém a engenharia enfrenta diversos desafios, como o do legado, que se propõe a dar suporte e atualização dos *softwares* atuais, procurando não aumentar os custos e manter serviços essenciais. Mais um desafio que podemos citar é o da heterogeneidade, que se refere a desenvolver recursos para construir *softwares* mais confiáveis e com maior flexibilidade para atender aos mais diversos tipos de equipamentos e métodos. Mais um aspecto desse contexto seria o desafio do fornecimento, que procura reduzir substancialmente o tempo que é gasto na criação e implantação do *software* sem prejudicar a qualidade.

2.3.1 Engenharia de Software Educacional

Confeccionar um *software* educacional com precisão e clareza tem sido um grande estímulo para a engenharia de *software*, pois antes de tudo precisa-se de um propósito educacional pedagogicamente apropriado e saber se transmite a mensagem com clareza para desenvolver inclusive a autoaprendizagem. Segundo Costa (2017), um *software* quando empregado como instrumento no processo de ensino-aprendizagem, os conceitos adquiridos previamente tornam-se importantes para novas construções cognitivas.

A engenharia de *software* educacional tem um público diversificado e necessita de conhecimento em diversas áreas e não só da computação, pois elas são fundamentais para a produção do *software* educacional. Para que todo o processo ocorra da melhor forma possível, vários profissionais como professores, psicólogos, psicopedagogos e tantos outros deixam sua contribuição para o *software*. E mesmo assim essa área ainda enfrenta uma deficiência de ferramentas para modelagem permitindo que todos os interessados possam compreender o que foi pensado e sugerido.

Costa (2017) ainda complementa afirmando que em muitos casos, para que haja um melhor aproveitamento dos temas trabalhados, a utilização de um ambiente virtual específico tende a desenvolver um papel muito importante didático e cognitivamente, pois se for visto da maneira mais apropriada impulsiona novas formas de aprender com mais dinâmica e eficiência.

É necessário ter arquiteturas de interoperabilidade e adaptar a engenharia as metodologias mais ágeis para um ganho expressivo na criação do *software* educacional. Também é essencial a criação de padrões de projetos para o segmento educacional e de modelos com padrões arquiteturais para domínios específicos. E não diferente de tudo isso, ainda é preciso determinar os moldes para avaliação do aprendizado com recursos humanos devidamente qualificados.

O engenheiro de *softwares* educativos precisa saber que diferente de outros programas comerciais esse tipo de projeto tem métodos previamente definidos que se transformam em operações. Pois um sistema educativo, determina conhecimentos utilizando uma metodologia eficiente de comunicação e assimilação do conteúdo proposto. Então é visto que o especialista em *softwares* educativos lida muito mais com os aspectos subjetivos da educação considerando a essência do material até as formas mais adequadas para a compreensão do conhecimento pelo usuário final.

2.4 METODOLOGIAS ÁGEIS

Produzir e entregar um *software* dentro dos prazos e com as metas estipuladas é sempre um desafio que muitas vezes não é cumprido, pois a formalidade nos modelos tradicionais de processo do programa atrapalha o cumprimento desses objetivos. Na contramão dessas situações podemos citar as metodologias ágeis que trazem o desenvolvimento de *software* de forma mais rápida, e buscando a mesma qualidade das metodologias tradicional. O Movimento Ágil se tornou forte em 2001, ano em que aconteceu a publicação do Manifesto Ágil de Desenvolvimento de *Software*, onde um grupo de especialistas se reuniu afim de criar ações para desenvolvimento de *software* de forma mais leve e humanizada, com maior rapidez e que fosse baseado nas pessoas. Esse manifesto que pouco tempo depois se tornou uma aliança ágil elencou princípios, boas práticas e valores que são usados até hoje por diversos profissionais adeptos dessa metodologia.

Para tanto pode-se observar que as metodologias ágeis contribuem consideravelmente para o desenvolvimento de *software* e com o segmento educacional não é diferente. Para entender melhor esse processo podemos lembrar os 12 princípios que norteiam o emprego de métodos ágeis

| |
|---|
| 1. A prioridade é satisfazer o cliente por meio de entrega contínua de <i>software</i> com valor agregado; |
| 2. Modificações de requisitos são bem-vindas, mesmo que sejam feitas no final do desenvolvimento. |
| 3. Entrega frequente de <i>softwares</i> funcionando, de preferência no menor espaço de tempo. |
| 4. Executivos e desenvolvedores e clientes trabalham juntos durante o projeto. |
| 5. A construção de projetos deve acontecer com pessoas motivadas e que tenham compromisso com o trabalho realizado. |
| 6. A comunicação entre a todos deve ser efetiva e acontecer cara a cara dentro da equipe de desenvolvimento. |
| 7. Se o <i>software</i> está funcionando este é o melhor cálculo de progresso. |
| 8. Manter passos constantes de desenvolvimento dentro de um ambiente sustentável. |
| 9. Atenção contínua a excelência técnica e ao bom design facilitam a agilidade. |
| 10. Simplicidade é fundamental. |
| 11. Equipes bem organizadas desenvolvem as melhores arquiteturas, requisitos e projetos. |
| 12. Refletir sobre eficácia, adaptando e ajustando adequadamente a equipe. |

Quadro 1: Princípios do Manifesto Ágil**Fonte:** Elaborado pela autora adaptado de Manifesto Ágil (2001).

Analisando as metodologias ágeis mais comumente usadas podemos citar o *Extreme Programming (XP)* que é focado na economia de recursos, na qualidade do produto e na rapidez para resolução de problemas. Também tem seus comportamentos baseados na filosofia ágil para garantir maior integração e desempenho. *Feed O FeatureDrivenDevelopment (FDD)* também é um método ágil que enfatiza as funcionalidades do sistema tendo o planejamento incremental como arma para agilizar o seu desenvolvimento.

Já o *Microsoft Solutions Framework (MSF)* é sem dúvida uma das metodologias ágeis mais utilizadas pois tem como premissa o desenvolvimento de soluções tecnológicas com o menor número de equipes com menos riscos e busca de qualificação do produto final. Ainda nas metodologias ágeis temos o *Dynamic System DevelopmentModel (DSDM)* que é muito antigo e diversificado dos outros métodos ágeis pois lida na maioria das vezes com prazos muito curtos e orçamento pré-determinado. O Scrum é a metodologia ágil mais utilizada atualmente, principalmente por carregar a característica da integração que pode ser empregada tanto na construção do *software* como nas relações de trabalho

2.4.1 Ypeduc

Empregado para a produção de *software* em meio acadêmico, o *easYProcces (YP)* é um exemplo de metodologia ágil com processo iterativo e incremental, porém simplificado, foi desenvolvido por meio de um projeto da Universidade Federal de Campina Grande (PROGRAMA DE EDUCAÇÃO TUTORIAL, 2007), trazendo mais uma opção de desenvolvimento de forma ágil. Abrange projetos simples mas tendo como característica uma metodologia bem desenvolvida com processos bem determinados e de fácil implantação.

Por ser uma metodologia ágil e voltada para a esfera acadêmica e passível de adaptações o *easYProcess (YP)* foi adaptado para ser empregado na área de *software* educacional e segundo Medeiros (2012), foram desenvolvidas as mudanças necessárias para utilizar a metodologia na produção educacional atendendo as necessidades

buscadas. Abaixo, na Figura 2 é apresentado o processo de fluxo do YPEDUC, o YP adaptado ao cenário educacional.

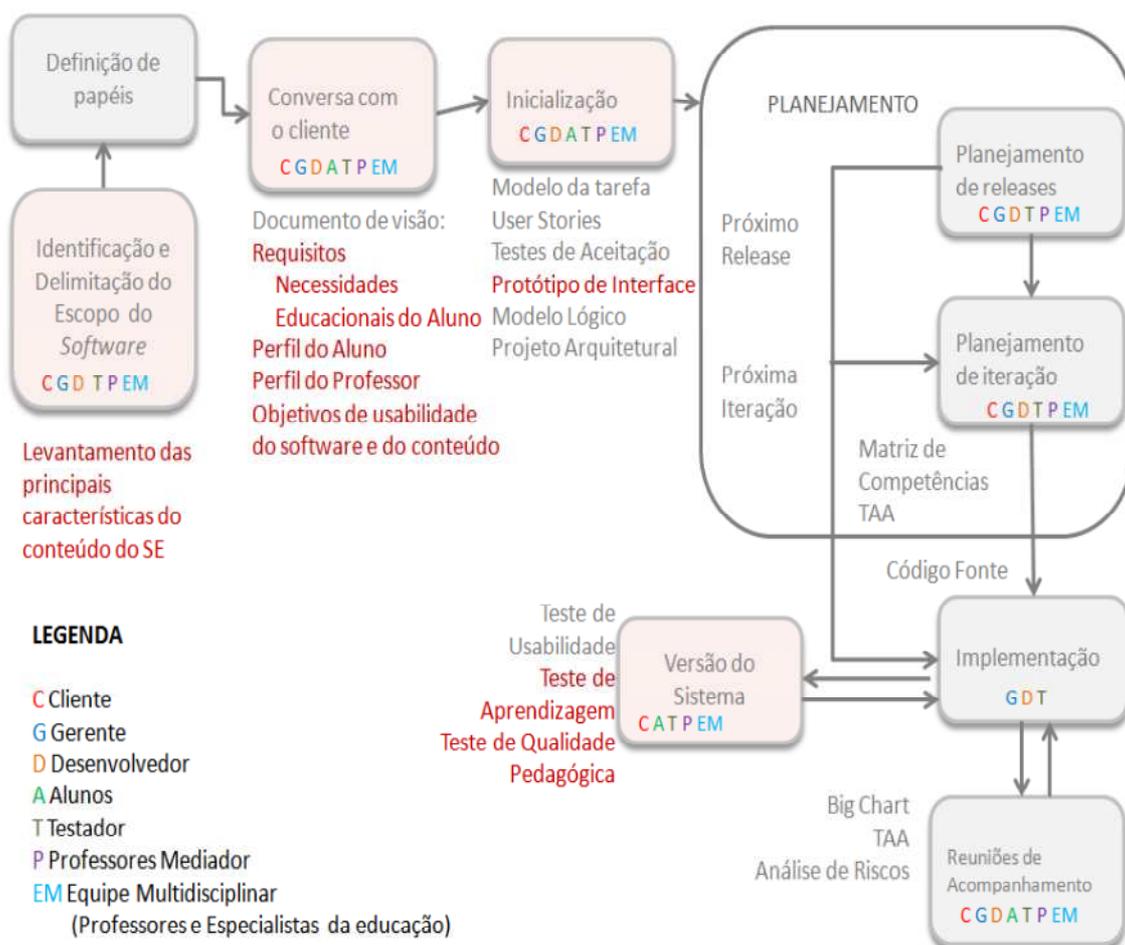


Figura 2: Fluxo do Processo YPEduc
Fonte: Medeiros (2012).

A seguir a autora descreve as mudanças que foram realizadas para a adaptação de todos os quesitos que tornaram o YPeduc uma metodologia educacional:

1. Na identificação do escopo do problema, além da assimilação do escopo do *software* foram elencadas todas as características referentes ao conteúdo abordado com os seus objetivos, define-se também o papel do *software* na aprendizagem, sua utilização e importância no processo. E toda a equipe participa da etapa.

2. Na especificação de papéis do YPeduc propõe-se a presença de: cliente, usuário (aluno), gerente, desenvolvedor, testador, professor (mediador) e equipe multidisciplinar (professor conteudista e pedagogos ou especialistas da educação).

3. Já na conversa com o cliente acontece a coleta de informações a respeito do sistema a ser criado, e participam da etapa o cliente, gerente, Desenvolvedor, Testador, Aluno, Professor Mediador e Equipe Multidisciplinar. A conversa com o cliente contempla os seguintes passos:

- Visão sobre o *Software* Educativo: Diante dessa conversa o cliente e os desenvolvedores encontram os pontos em comum a respeito do sistema, e é papel do desenvolvedor deve colher do cliente várias informações sobre o sistema e fazendo uma relação com suas funcionalidades (requisitos funcionais) e não funcionais, assim como perfil do usuário e objetivos de usabilidade e os respectivos testes, etc. A equipe multidisciplinar deve estar presente em toda a parte pedagógica do processo.
- Levantamento dos Requisitos Funcionais e Não Funcionais: A análise de requisitos em *softwares* educativos deve respeitar questões como público-alvo, contexto, conteúdo e avaliação.
- Levantamento do Perfil do Usuário/Aluno: Para o projeto de SE, no Perfil do Usuário deve ser estudado também o conhecimento prévio sobre o conteúdo abordado pelo *software* e diferente do que acontece em projetos de *softwares* comerciais.
- Levantamento do Perfil do Professor Mediador: O perfil do professor mediador traz uma noção do seu conhecimento sobre o conteúdo, experiência com tecnologias na educação, os principais desafios encontrados para transmitir o conteúdo abordado pelo SE, dentre outras.
- Levantamento dos Objetivos de Usabilidade: Na exposição dos objetivos de usabilidade leva-se em consideração não só a aprendizagem do sistema, mas

também o conteúdo educacional. É fundamental que professor e aluno também participem da produção dos objetivos de usabilidade.

4. Na inicialização começam as atividades de análise do sistema, onde participam da etapa o cliente, gerente, Desenvolvedor, Testador, Aluno, Professor Mediador e Equipe Multidisciplinar. Nesta etapa começa a construção de um modelo de tarefa que serve como base para o protótipo de interface que tem como função receber o *feedback* do cliente nas fases iniciais de implementação e listar todas as *UserStories* do sistema (e os Testes de Aceitação) a fim de desenvolver uma arquitetura, um modelo lógico de dados com capacidade para receber as mudanças e que sejam estáveis o suficiente para que riscos sejam minimizados. Dentro da inicialização encontram-se os seguintes passos:

- Modelagem da Tarefa: A construção do modelo da tarefa antecede a análise da tarefa, que é um estudo com a finalidade de melhorar a compreensão dos desenvolvedores com relação à natureza da tarefa, sua composição e a ordem de realização. São considerados os aspectos que ligam os objetivos de usabilidade, o perfil do usuário e todas as características da tarefa durante a sua análise;
- Levantamento das *UserStories* e Testes de Aceitação: nesta etapa o cliente faz suas especificações para definir o que o sistema deve conter e fazer. O uso no contexto educacional necessita ser considerado e respeitado para a produção das *UserStories*. Cada *UserStory* deve estar associada no mínimo a um Teste de Aceitação;
- Geração do Protótipo da Interface: Em um SE, na fase de prototipação da interface leva-se em consideração às múltiplas inteligências e o perfil dos alunos, a fim de atingir o máximo de alunos possível com o aprendizado intuitivo. Para a produção são empregados métodos de design centrado no usuário e que reúnam questões pertinentes a modelos cognitivos do processamento humano;
- Elaboração do Projeto Arquitetural: Aqui acontece a descrição do funcionamento do sistema num alto nível de abstração. Tem uma importância muito grande para explicar como as partes do sistema interagem entre si ou com outros sistemas;
- Geração do Modelo Lógico de Dados: Se o sistema conter banco de dados, o desenvolvedor deve construir o modelo lógico de dados capaz de representar a estrutura lógica do banco de dados.

5. Planejamento (*Release* e Iteração). Quando são concluídas as atividades de inicialização aí tem início o planejamento do projeto e a equipe já deve ter determinado o tempo disponível para o seu desenvolvimento, e a partir daí com o cliente, definir o número de *releases* e iterações necessárias para a conclusão do mesmo.

Pequenos releases são efetuados para ter uma maior relação entre a equipe de desenvolvimento e o cliente, sendo assim, as *UserStories* alocadas em cada release só podem ser consideradas como finalizadas após a realização dos testes de aceitação.

O plano da iteração quebra as *UserStories* em atividades menores (quando é preciso) e coloca os desenvolvedores como responsáveis pela realização de cada atividade. A atribuição da tarefa para o desenvolvedor pode ser feita tendo a ajuda da matriz de competências que é uma tabela simplista que diz quais as capacidades dos membros da equipe de desenvolvimento e o tempo dedicado por cada um no projeto. Quem deve participar da etapa devem ser o cliente, o gerente, desenvolvedor, testador, professor mediador e equipe multidisciplinar.

6. Já na implementação acontece a execução das atividades determinadas no plano de iteração, e o principal artefato é o código do sistema. E para que o resultado seja uma boa codificação é preciso que a equipe de desenvolvimento conceitue as seguintes práticas: Integração Contínua, Boas Práticas de Codificação, Propriedade Coletiva de Código, Testes e Pequenos *Releases*. Devem participar desta etapa o gerente, Desenvolvedor e Testador.

7. Na versão do produto acontece apuração de todos os requisitos do sistema, para saber se a implementação está correta. Devem participar desta etapa o cliente, Testador, Aluno, Professor Mediador e Equipe Multidisciplinar. Nesta fase, é necessário que além dos Testes de Usabilidade também ocorram os Testes de Aprendizagem e de Qualidade Pedagógica, criado pela equipe multidisciplinar para avaliar a aprendizagem dos alunos e o caráter pedagógico do *software*.

- Testes de Usabilidade: Averigua se os objetivos de usabilidade detalhados pelo cliente e pelo usuário foram realizados, além de verificar como está acontecendo o processo de interação entre o usuário e o sistema. Os campos de teste de usabilidade devem procurar apreciar o desempenho de usuários típicos na realização de tarefas comuns.
- Testes de Aprendizagem: Demonstra se os alunos – usuários do SE estão de fato aprendendo o conteúdo substancial do aplicativo, ou seja, é analisado o que o aluno absorve de conhecimento após a sua aplicação. Inúmeros métodos de

avaliação podem ser aplicados para o Teste de Aprendizagem, como provas, questionários, mini testes, dinâmicas, dentre outras.

- Testes de Qualidade Pedagógica do *Software* Educativo: Aqui são verificados os aspectos pedagógicos (facilidade no acesso às informações, adequação a faixa etária, clareza nas informações, tipos de exercícios), do conteúdo (fidelidade ao objeto, coerência de apresentação do conteúdo, correção dos exercícios, organização dos conteúdos, promoção da criatividade e motivação dos usuários) e aspectos gerais (alcança os objetos propostos, contribui para a aprendizagem dos conteúdos apresentados, preço compatível). Para o Teste de Qualidade Pedagógica existem alguns métodos estruturados indicados para a avaliação de qualidade de *softwares* educacionais, dentre eles a TICESE - Técnica de Inspeção Ergonômica de *Software* Educacional (GAMEZ, 1998) e a Ficha de Avaliação de Software Educativo (Metodologia de Avaliação de *Software* Educativo Adaptado LECT/ USP).

2.5 INFORMÁTICA EM SAÚDE

A informática foi criada para suprir as carências das áreas de guerra e militares, só que o que se observa é que sua utilização se estendeu para as mais diversas esferas do conhecimento, e sua aplicação na saúde é real e acaba provocando a aproximação de diversos saberes. Seja no processamento e estudo de dados que fazem um levantamento da prática epidemiológica da população com análises estatísticas ou na pesquisa e interpretação dos indicadores de letalidade ou em outras subáreas, a informática chega como uma nova ciência que tem o poder de transformar realidades e reorganizar os sistemas de informação a respeito da saúde.

É pertinente admitir que o profissional da área da saúde, em função das necessidades impostas pelo advento tecnológico contemporâneo, especialmente nesse setor, desenvolva competências e saberes relativos a um “pensar e agir” que inclua as TICs no intuito de enriquecer e ampliar sua prática profissional, sua educação permanente e sua participação social nos campos especiais em que vier a atuar. (CARDOSO, 2008, p. 284)

No campo da saúde a informação que é tratada e coletada corretamente e de forma atualizada serve de base para determinação de várias ações, não só para o controle de doenças mas para uma melhora da qualidade de vida da população, atuando como ferramenta de controle social e nessa ótica também é possível qualificar os serviços da saúde.

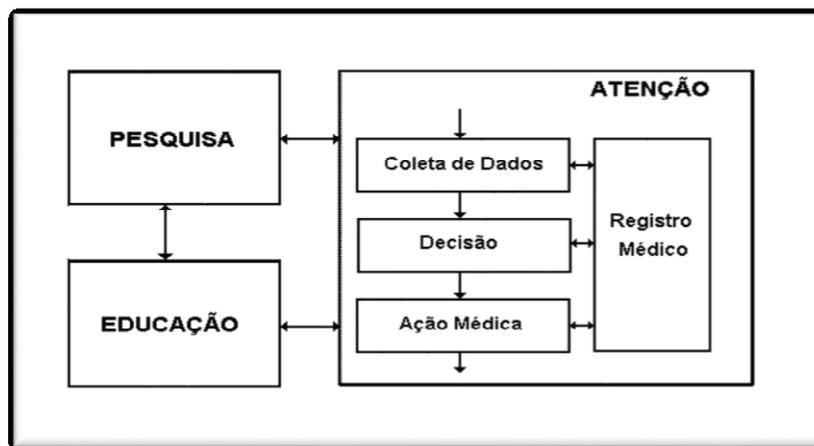


Figura 3: Processo e Informação em Medicina
Fonte:Sabbatini (1994).

O campo de atuação da informática na medicina inclui registro médico computadorizado, processamento digital de sinais biológicos e de imagens médicas, sistemas de apoio a decisão, terapias e aplicações no ensino médico, assim como epidemiologia e bioestatística promovem um acesso maior da informação em saúde e possibilita que a informática chegue a clínica ao consultório e ao hospital.

Um dos maiores exemplos de informática em saúde foi a criação do Departamento de Informática do Sistema Único de Saúde (Datasus) que surgiu para o controle e processamento de contas da saúde. E tem competências e responsabilidades de abastecer os órgãos do SUS (Sistema Único de Saúde) com suporte a informática para planejamento, operação e controle. Em quase 25 anos de atuação, o Datasus já desenvolveu mais de 200 sistemas que auxiliam diretamente o Ministério da Saúde no processo de construção e fortalecimento do SUS. No momento presente, o departamento é um grande provedor de soluções de *software* para as secretarias estaduais e municipais de saúde, sempre adaptando seus sistemas às necessidades dos gestores e incorporando novas tecnologias, na medida em que a descentralização da gestão torna-se mais concreta. (DATASUS, 2017).

O uso de computadores de mão (*palmtop*), de sistemas com conexão sem fio (*wireless*) e dispositivos móveis, assim como a disseminação dos celulares habilitados com diversas funções como os smartphones trouxe uma transformação tecnológica considerável nas últimas décadas, inclusive com a criação de vários aplicativos destinados as mais diversas áreas e de certa forma dão muita assistência à saúde, seja para os profissionais da área ou para os usuários comuns.

2.5.1 Promoção da Saúde Vocal do Professor

Desde de o início do século a Fonoaudiologia em nosso país, possui predominância da ação clínica e da atenção individual em detrimento de ações preventivas/coletivas (SERVILHA *In*: FERREIRA, LOPES e LIMONGI, 2004).

Expandiu-se em 1990, a inserção de fonoaudiólogos em contextos coletivos, direcionando uma atenção especial à área de Voz Profissional. Novas necessidades, tendências do mercado e avanços tecnológicos colaboraram para o surgimento da valorização de profissões que exigiam o uso da voz para sua concretização/viabilização, os denominados profissionais da voz (SERVILHA *In*: FERREIRA, LOPES e LIMONGI, 2004).

Verifica-se em alguns estudos que na formação do professor não há orientações sobre os cuidados com a saúde vocal o que como consequência pode favorecer ao despreparo do professor frente a demanda vocal no decorrer do exercício do trabalho (BEHLAU, NAGANO e DRAGONE, 2004).

Devido à alta prevalência de problemas vocais e um alto número de professores no país, é de suma importância, o estudo e o desenvolvimento de ações preventivas enfocando a importância da conscientização das pessoas a respeito dos cuidados com a voz.

2.6 DISFONIA

A disfonia consiste numa perturbação da comunicação verbal oral e que representa qualquer dificuldade ou alteração na emissão vocal que impede a produção natural da voz (BEHLAU, 2004; MARÇAL e PERES, 2011).

Os fatores de risco para desenvolver patologia vocal incluem as características individuais ambientais as doenças respiratórias, a medicação, o consumo excessivo de substâncias o refluxo gastro esofágico e a atividade profissional são alguns dos seus fatores causais (CONSTANTINO e GUIMARÃES, 2005).

A disfonia dificulta a produção natural da voz e afeta negativamente a qualidade vocal, alterando os parâmetros vocais acústicos afirmam os autores Constantino e Guimarães (2005). A avaliação qualidade vocal e disfonia pode ser realizada por observação e questionários ou por gravação e análise de emissão vocal através de *softwares* que podem auxiliar este processo e que podem ser gratuitos como *Audacity*,

EMU *Speech DatabaseSystem*, *Praat*, *SpeechFilingSystem* (SFS) entre outros ou pagos como é o caso dos *Dr. Speech*, *VideoVoiceSpeech*, *TrainingSystem*, *Voxmetria* e *Vocalgrama* etc. (MENDES, et al 2012).

2.7 ALTERAÇÕES VOCAIS NOS DOCENTES

Os profissionais da voz incluem os docentes, os terapeutas da fala, advogados, relações públicas, vendedores, etc. Estes profissionais precisam de uma boa qualidade vocal para alcançar um bom desempenho profissional.

Os docentes apresentam um risco acrescido para desenvolver problemas vocais, por exercer o seu a sua atividade de trabalho pelo uso da voz. Em alguns estudos realizados verificou-se que a maior parte dos docentes possui poucos conhecimentos sobre os mecanismos envolvidos na fonação (BEHLAU e PONTES, 1995; PINHO, 2003).

Alguns dos principais problemas ambientais e comportamentos de mau uso e abuso que provocam uma alteração vocal e indicam a qualidade da comunicação em sala de aula, e são responsáveis pelas queixas e sintomas que podem ser indicadores de patologias laríngeas (CHOI-CARDIM, BEHLAU e ZAMBON., 2010). Sendo assim é necessário intervir, através da realização ações para consciencializar a classe docente sobre a importância da voz.

3 MINIPROJETO PARA O DESENVOLVIMENTO DO SOFTWARE EDUCATIVO: YVOICE

O escopo de um projeto descreve todos os seus produtos, os serviços necessários para realizá-los e os resultados esperados. E aqui foi o primeiro objeto confeccionado para o desenvolvimento do *software* YVOICE.

| IDENTIFICAÇÃO DO ESCOPO DO PROBLEMA |
|--|
| O sistema a ser criado tem uma conotação educacional e o seu objetivo principal é chamar atenção e ao mesmo tempo incentivar os professores a conhecer mais sobre a utilização de sua voz, fazendo com que através de um aplicativo interativo e dinâmico os professores entendam a importância dos cuidados para a preservação vocal. |

Quadro 2: Escopo do Problema

Fonte: Elaborado pela autora (2017).

Em seguida, foi realizada a definição de papéis com o intuito de dividir tarefas e responsabilidades do projeto, este artefato está sendo mostrado no quadro 3.

| PAPÉIS NO PROCESSO YEDUC | |
|---------------------------------|---|
| Tarciana Abrantes | Cliente |
| Tarciana Abrantes | Gerente, Testador, Professor (conteúdistas), Pedagogo e Usuário (aluno), e Professor Mediador |
| Tarciana Abrantes | Desenvolvedor |

Quadro 3: Definição de Papéis

Fonte: Elaborado pela autora (2017).

O documento de visão determina o escopo de alto nível e a finalidade de um programa, produto ou projeto. É uma solução proposta e os recursos de alto nível do produto ajudam a estabelecer expectativas e a reduzir riscos. Então após a divisão dos papéis ocorre a conversa com o cliente e a partir dele surge a descrição mais detalhada do sistema como mostra o quadro 4. A definição dos requisitos funcionais e não funcionais do sistema está demonstrada no quadro 5.

VISÃO SOBRE O APLICATIVO YVOICE

As tecnologias e dispositivos móveis, tais como celulares, *smartphones* e *tablets* evoluem de maneira cada vez mais rápida quando comparados, por exemplo, aos computadores pessoais e podem ser usadas como ferramentas para a promoção da saúde utilizando aplicativos.

Dentre os profissionais da voz, o professor é um dos mais susceptíveis a ter um distúrbio vocal em decorrência de diversos fatores relacionados ao ambiente de trabalho que se caracteriza muitas vezes: por ter uma jornada de tarefas muito longa, demanda vocal excessiva, sala de aula com acústica insatisfatória, ruído interno e externo, entre outros. O distúrbio de voz relacionado ao trabalho (DVRT) é uma doença ocupacional, multifatorial, resultante de fatores ambientais e/ou organizacionais do trabalho, condições vocais inadequadas e fatores predisponentes do indivíduo, que atinge principalmente trabalhadores que usam a voz como principal instrumento de trabalho.

Sabe-se que os distúrbios de voz podem ocasionar diversos impactos na atividade profissional do professor: impacto na expressão vocal e o impacto emocional, que causa forte estresse e ansiedade; colocando em risco a carreira e a sobrevivência do trabalhador, daí surgiu a necessidade de um *software* educacional que pudesse auxiliar esses profissionais e orientá-los no tocante a educação vocal, adquirindo assim uma melhoria na saúde.

O *software* que se apresentará na forma de App deve conter uma dinâmica educacional que envolva o usuário, sugere-se uma interface amigável e um design que prenda o cliente dando-lhe consciência da real importância do aplicativo. Na tela inicial se encontra uma interface amigável e que tem como intuito despertar a curiosidade do usuário, com o nome do aplicativo (YVOICE) e o motivo pelo qual ele está sendo criado, que é a promoção da saúde vocal, nesta tela encontra-se também uma imagem sobre a voz, para ilustrar e chamar a atenção. O fundo da tela será colorido e as letras em fonte maior e destacadas, que contam como recurso gráfico para atrair a atenção do usuário, e no canto

direito da tela se encontra uma seta com o nome “avançar” que induz o leitor a navegar para a próxima tela do aplicativo.

A proposta da tela 2 começa com o conhecimento básico sobre **anatomia** do aparelho vocal, que é fundamental para o raciocínio e assimilação dos tópicos abordados durante a utilização do App e também o melhor entendimento da produção da voz implica em uma maior adesão destes profissionais a ações propostas. Portanto uma seção explicativa sobre aparelho fonador com ilustrações e explicação escrita, além de vídeos e imagens se faz necessária para que antes de tudo o usuário saiba como a voz é produzida. Para tanto, faz-se necessária a presença de um dicionário ou de um glossário com a terminologia adequada para um melhor entendimento dos termos relacionados a área. A cor do fundo desta e das demais telas será na cor branca para facilitar a leitura, e letras pretas para se destacar, também se encontra no canto direito o botão ‘inimigos’ que conduzirá o usuário para a próxima tela, e no canto esquerdo está o botão ‘voltar’ que o retorna para a tela inicial do App, esta terminologia se aplica a todas as telas do aplicativo.

Em seguida o App deve ter uma sessão intitulada ‘os inimigos da voz’ com uma breve explicação sobre os problemas que acontecem com o aparelho vocal caso o usuário utilize a voz sem a devida orientação. Em seguida é listado cada inimigo da voz e o usuário deve clicar em cima do nome para descobrir as implicações de cada um, e encontra-se exemplos, imagens e vídeos que reforcem essas implicações.

A partir daí começam as dicas sobre os cuidados com a voz – orientações sobre a produção vocal através de imagens, ilustrações e demonstrações. Para atingir tal necessidade, tornando o uso do sistema mais agradável, sugere-se também a utilização de vídeos. Nesta tela encontra-se uma relação de cuidados com a voz e o usuário deve clicar em cima de cada uma delas para encontrar uma explicação escrita com imagens e vídeos. Adicionalmente, como uma maneira de qualificar as dicas e tornar possível a interação entre os interessados no sistema, espera-se a disponibilidade de uma ferramenta que promova interação, tal qual um fórum aberto às discussões, onde os usuários podem interagir e discutir os mais diversos temas relacionados a voz.

Uma outra requisição feita pelo cliente, versa sobre “Voz saudável, comunicação em dia!”, uma área contendo esclarecimentos com os mitos e verdades sobre a voz, além de orientações sobre higiene vocal através de imagens e vídeos e de uma seção de FAQ’s (questões mais frequentemente realizadas) sobre o assunto.

Ainda se faz necessária uma seção intitulada “Voz profissional”, onde devem ser consideradas orientações sobre o papel da voz em diversas profissões, exposição de imagens e explicação escrita sobre a importância dos cuidados com a voz para profissionais da voz como professores, operadores de telemarketing, atores, cantores, radialistas e etc. Por fim, na última tela encontra-se um mecanismo de busca sobre os profissionais da área

de saúde vocal, incluindo fonoaudiólogos, otorrinolaringologistas etc. Esta busca é local, onde o usuário se encontrar basta digitar o nome da cidade que deseja e terá um resultado de todos os profissionais de saúde vocal mais próximos de sua localização, com endereço, telefone e especialidade.

Quadro 4: Descrição do Sistema

Fonte: Elaborado pela autora (2017).

| REQUISITOS FUNCIONAIS | |
|-----------------------------|---|
| Componente Instrucional | Disponibilizar glossário; Informacional; Publicizar orientações; Disponibilizar fórum; Mensagens de incentivo. |
| Componente de Profissionais | Cadastro de médicos e fonoaudiólogos; Busca por médicos e fonoaudiólogos. |
| Componente de Perguntas | Cadastro de perguntas; Cadastro de respostas; Divulgar as FAQ's; |
| REQUISITOS NÃO FUNCIONAIS | |
| Facilidade de Uso | Cor de fundo e da letra contrastante; Telas simples e interação intuitiva; Disponibilização de navegação; Guia de ajuda. |
| Desempenho | Pelo fato do App ser relativamente simples o mesmo não apresenta nenhuma restrição de desempenho. |
| Portabilidade | móvel android; ios |
| Entrega | Dois meses, |
| Implementação | Linguagem <i>Javascript</i> ; Método iterativo e incremental. |
| Padrões | Não se aplica ao sistema. |
| Espaço | Otimizar o tamanho final do apk |
| Confiabilidade | Pouca propensão a erros. |
| Interoperabilidade | Não se aplica ao sistema. |
| Ético | <i>Software</i> de interesse público. |
| Privacidade | Respeito aos direitos autorais. |
| Segurança | Não se aplica ao sistema. |
| Público-alvo | Professores e diversos profissionais que tem a voz como instrumento de trabalho. |
| Contexto | O <i>software</i> poderá ser utilizado em qualquer lugar, pois se apresenta na forma de App. |
| Conteúdo | Voz. |
| Avaliação | Ferramentas de análise, <i>flurry</i> |

Quadro 5: Definição dos Requisitos

Fonte: Elaborado pela autora (2017).

O perfil do usuário e dos profissionais da área são gerados a partir da primeira conversa com o cliente. Tais artefatos seguem nos quadros 6 e 7, respectivamente. Para coletar dados para o perfil do professor, foi necessário que o mesmo respondesse um questionário adaptado do Laboratório de Interfaces Homem-Máquina, UFCG que é usado para levantamento do perfil do usuário.

| FORMULÁRIO PARA O LEVANTAMENTO DO PERFIL DO PROFESSOR | |
|--|---|
| Características do usuário, escolhidas pelo projetista, de acordo com a relevância, para o projeto. Levantamento baseado em: | |
| Fatos () Opinião do usuário () Dados medidos ou observados () Opinião do Usuário (X) | |
| Parte I – Características Gerais | |
| Faixa etária: 20 a 50 anos. Sexo: Feminino/ Masculino. Habilidades específicas necessárias para executar a tarefa: Níveis de percepção: Visão/ tato/ audição Habilidades motoras: Um mínimo possível de coordenação motora. Grau de instrução (ex. técnico, superior): médio/ superior. Função desempenhada: Aprendiz / aluno. Objetivos do aplicativo: Despertar os cuidados relacionados à voz. Motivações do usuário: Aprender sobre produção vocal, cuidados com a voz, doenças e problemas causados por mau uso vocal. Preferências (ex. uso do teclado, preenchimento do formulário): Celular e plataformas móveis. | |
| Parte II - Conhecimento Conceitual: necessário à execução das tarefas | |
| CONHECIMENTO SEMÂNTICO | |
| Nível de Experiência (noviço, experiente, especialista) | |
| Função: experiente | Método: Noviço |
| Tarefa: Noviço | Computadores/ celular/tablet: experiente |
| Ferramentas utilizadas na execução das tarefas (ou similares): experiente | |
| CONHECIMENTO SINTÁTICO | |
| Nível de experiência | |
| Uso de teclado e mouse: experiente | Uso de dispositivos especiais de interação: experiente |
| Uso de terminologia específica: experiente | |
| Parte III – Estilo Cognitivo | |
| Aprendizado (treinamentos): Através do guia de ajuda do App | |
| Capacidade de solucionar problemas (sozinho, com ajuda): Sozinho. | |
| Capacidade de reter o aprendizado (alta, média, baixa): Alta. | |
| PERSONALIDADE | |
| Nível de curiosidade (baixo, médio, elevado): Elevado. Nível de persistência (baixo, médio, elevado): Elevado. Nível de inovação (baixo, médio, elevado): Médio. Inovador () Conservador () Impulsivo (X) Reflexivo () | |

| |
|--|
| CONHECIMENTO PRÉVIO |
| <p>Quanto às telas, o usuário: Não reconhece nenhuma () Reconhece algumas () Reconhece Todas (X)</p> <p>Quanto as figuras e sons, o usuário: Não reconhece nenhum () Reconhece alguns () Reconhece Todos (X)</p> <p>Quanto ao grau de leitura do usuário: () Não sabe/Está aprendendo () Intermediário () Alfabetizado (X) Leitor Crítico</p> <p>O usuário terá aulas sobre o tema abordado antes de utilizar o software: Sim () Não (X)</p> <p>O usuário já apresentou algum tipo de curiosidade/interesse sobre o tema: Sim () Não (X)</p> <p>Qual o nível de conhecimento que o usuário possui sobre o tema: Baixo () Médio (X) Alto () Especialista ()</p> <p>O usuário será avaliado após a utilização do software: Sim () Não (X)</p> <p>Em geral, o usuário apresenta alguma dificuldade sobre o tema proposto conteúdo: Não () Sim (X) Qual: a maioria dos profissionais só se dá conta da importância dos cuidados com a voz quando já apresenta algum problema vocal.</p> |

Quadro 6: Perfil do Professor

Fonte: Formulário adaptado do Laboratório de Interfaces Homem-Máquina, UFCG (DEPARTAMENTO DE ENGENHARIA ELÉTRICA, 2003).

| |
|---|
| FORMULÁRIO PARA O LEVANTAMENTO DO PERFIL DO PROFISSIONAL DA SAÚDE |
| <p>Características do profissional, escolhidas pelo projetista, de acordo com a relevância, para o projeto. Levantamento baseado em: Fatos () Opinião do usuário () Dados medidos ou observados () Opinião do Profissional (X).</p> |
| Parte I – Características Gerais |
| <p>Qual a sua formação Acadêmica: Médico/ fonoaudiólogo/otorrinolaringologista Nível de experiência na área: Baixo () Médio () Alto (X) Fez algum curso de informática: Sim (X) Não () Qual o seu grau de instrução tecnológica: Baixo () Médio (X) Alto () Quanto costuma utilizar o computador/ Tablet/ Celular: Raramente () Semanalmente () Diariamente () Várias vezes por dia (X) Objetivos do aplicativo: Despertar os cuidados relacionados à voz/ Cadastrar profissionais Motivações do usuário: Aprender sobre produção vocal, cuidados com a voz, doenças e problemas causados por mau uso vocal e ter uma lista de profissionais para cuidar da voz através de um mecanismo de busca baseado na localização Preferências (ex. uso do teclado, preenchimento do formulário): Teclado e/ou mouse.</p> |
| Parte II - Conhecimento Conceitual: necessário à execução das tarefas |
| <p>CONHECIMENTO SEMÂNTICO Nível de Experiência (noviço, experiente, especialista)</p> |
| <p>Função: Experiente Método: Experiente Tarefa: Experiente Computadores: Experiente Ferramentas utilizadas na execução das tarefas (ou similares): Experiente</p> |
| <p>CONHECIMENTO SINTÁTICO Nível de experiência</p> |
| <p>Uso de teclado e mouse: Experiente Uso de dispositivos especiais de interação: Experiente Uso de terminologia específica: Experiente</p> |
| Parte III – Estilo Cognitivo |

| | | | |
|---|---------------------|-------------------------|--------------------|
| PERSONALIDADE | | | |
| Nível de curiosidade (baixo, médio, elevado): Elevado. | | | |
| Nível de persistência (baixo, médio, elevado): Elevado. | | | |
| Nível de inovação (baixo, médio, elevado): Médio. | | | |
| Inovador () Conservador () Impulsivo () Reflexivo (X) | | | |
| CONTEÚDO | | | |
| Qual o seu nível de conhecimento sobre o tema: | | | |
| Baixo () | Médio () | Alto () | Especialista (X) |
| O usuário já teve alguma orientação sobre este tema: Sim (X) Não () | | | |
| Em geral, o usuário apresenta alguma dificuldade sobre o tema proposto conteúdo: | | | |
| Não (X) Sim () | | | |
| Qual tipo de Avaliação você realizará após a utilização do <i>software</i>: | | | |
| Prova () | Perguntas Orais () | Conversa informal (X) | Seminários () |
| Questionários () | Nenhuma () | Outras () _____ | |

Quadro 7: Formulário para o levantamento do Perfil Profissional da Saúde

Fonte: Formulário adaptado do Laboratório de Interfaces Homem-Máquina, UFCG (DEPARTAMENTO DE ENGENHARIA ELÉTRICA, 2003).

Outra peça resultante desta etapa do projeto são os objetivos de usabilidade do *software* YVOICE, os objetivos de usabilidade e a mensuração destes estão apresentados no quadro 8.

| OBJETIVOS DE USABILIDADE DO YVOICE | |
|---|---|
| OBJETIVOS | MENSURAÇÃO |
| Reduzir a taxa de erros | Tentar usar mensagens claras. |
| Facilitar o aprendizado | Cores e imagens intuitivas. |
| Adequar o conteúdo | Respeito com o nível de conhecimento do usuário. |
| Aumentar a satisfação subjetiva do usuário | Consultar usuário após 1ª impressão com o sistema a fim de otimiza-lo em novas versões. |
| Manter a clareza na estrutura | Guia de usabilidade de fácil entendimento e presente no próprio App. |
| Ser atrativo ao usuário | Consultar usuário com relação a interface gráfica. |
| Possuir telas simples | Interface intuitiva e de fácil interação. |
| Manter consistência nas sequências de ações necessárias para realizar uma tarefa | Simples acesso e manipulação. |
| Disponibilizar boa documentação | Documentação simples incorporada no próprio <i>software</i> . |
| Reduzir a dificuldade de aprendizagem do usuário com relação ao conteúdo abordado | Exploração do conteúdo de forma simples e objetiva. |

| | |
|---|--|
| Sanar as dúvidas que surgirem quanto ao conteúdo do <i>software</i> | Na ajuda deve conter uma explicação sucinta sobre o uso do App |
|---|--|

Quadro 8: Objetivos de Usabilidade do Yvoice

Fonte:Elaborado pela autora (2017).

A próxima fase do projeto é a Inicialização que é a fase de proposta do projeto, onde o mesmo é avaliado com relação ao seu real objetivo, a sua prioridade e impacto com relação a organização. Aqui a equipe já deve ter ciência da ideia geral do projeto e do sistema que está em desenvolvimento, pois é agora que começam as atividades de análise do sistema. Os artificios que foram criados nesta fase foram: Modelagem da tarefa, *UserStories* e Testes de Aceitação, Protótipo de Interface, Projeto Arquitetural e Modelo Lógico de Dados. A modelagem da tarefa foi realizada pelo desenvolvedor através da Ferramenta para Modelagem de Tarefas ITAOS (PETRÔNIO et al, 2010) e está exposta no quadro 9.

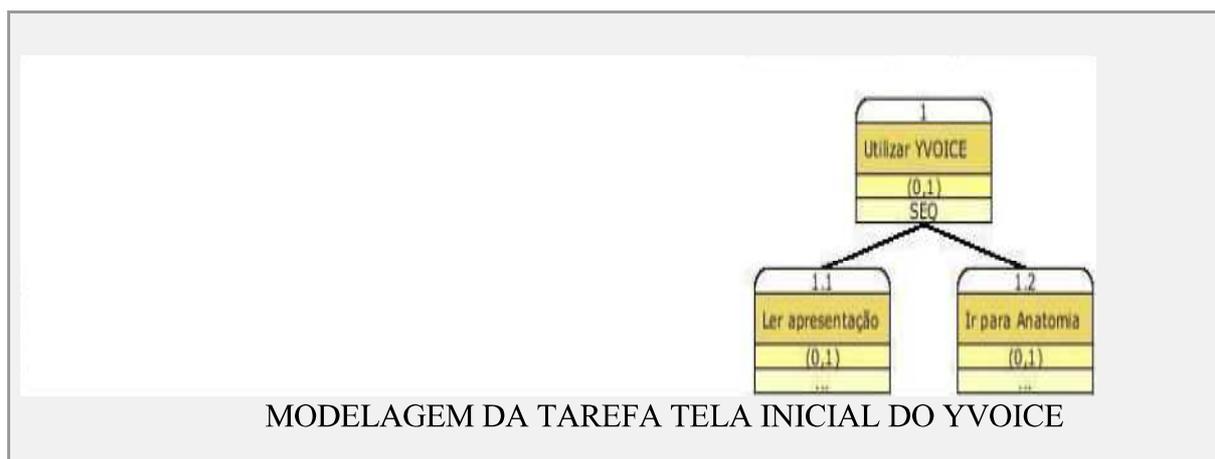


Figura 4: Modelagem da Tela 1

Fonte: Elaborado pela autora (2017).

Na modelagem da Tela 1 mostra-se a utilização da tela inicial do aplicativo, onde é possível ler a apresentação da ferramenta, ou avançar para a próxima tela.

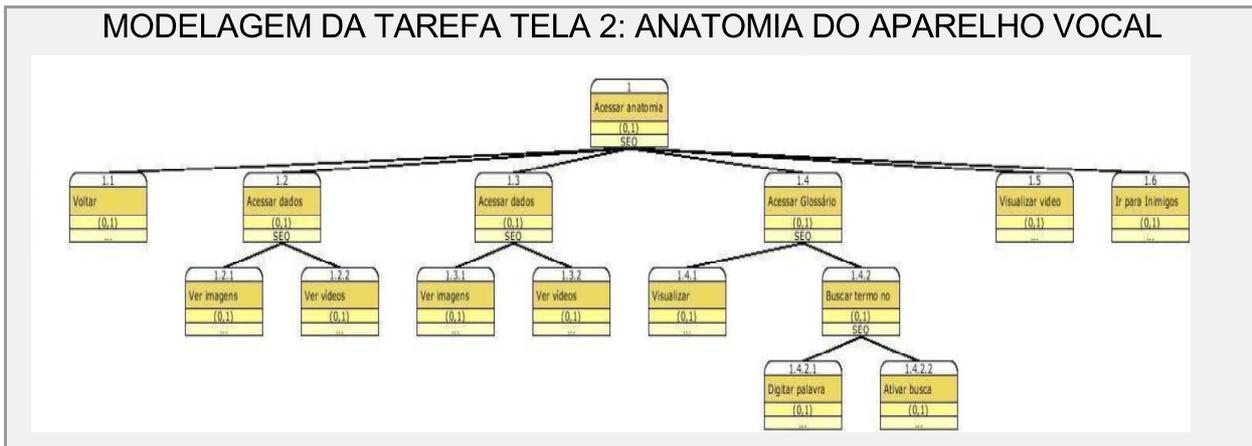


Figura 5: Modelagem da tela 2
Fonte: Elaborado pela autora (2017).

Na modelagem da tela 2 é possível acessar os dados sobre anatomia do aparelho vocal, acessar o glossário, voltar ou avançar para a próxima tela.

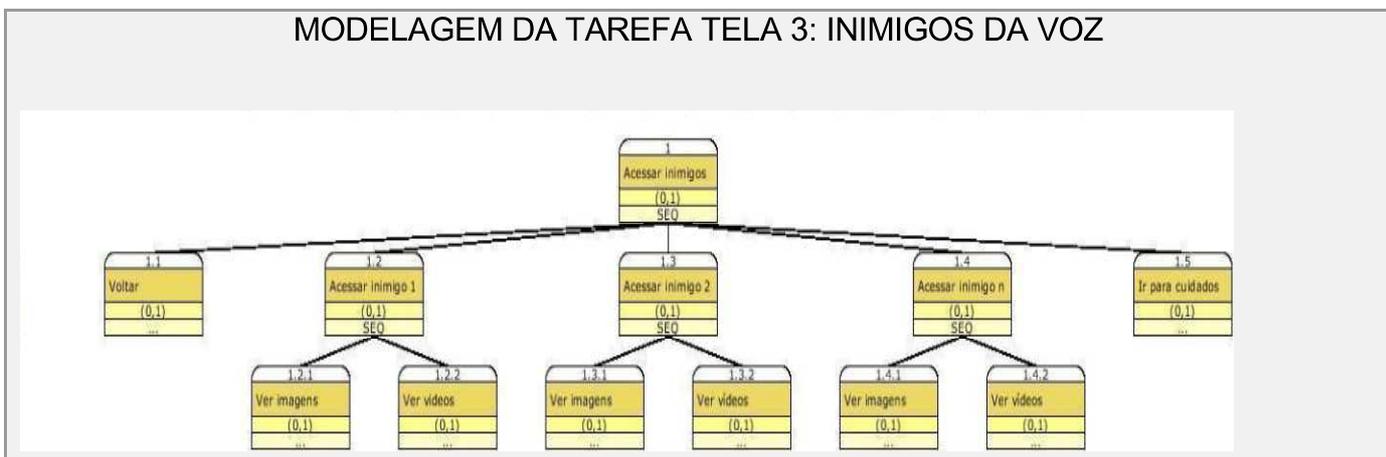


Figura 6: Modelagem da tela 3
Fonte: Elaborado pela autora (2017).

Na tela 3 pode-se ter acesso a todo o conteúdo sobre os inimigos da voz, ou avançar para a próxima tela.

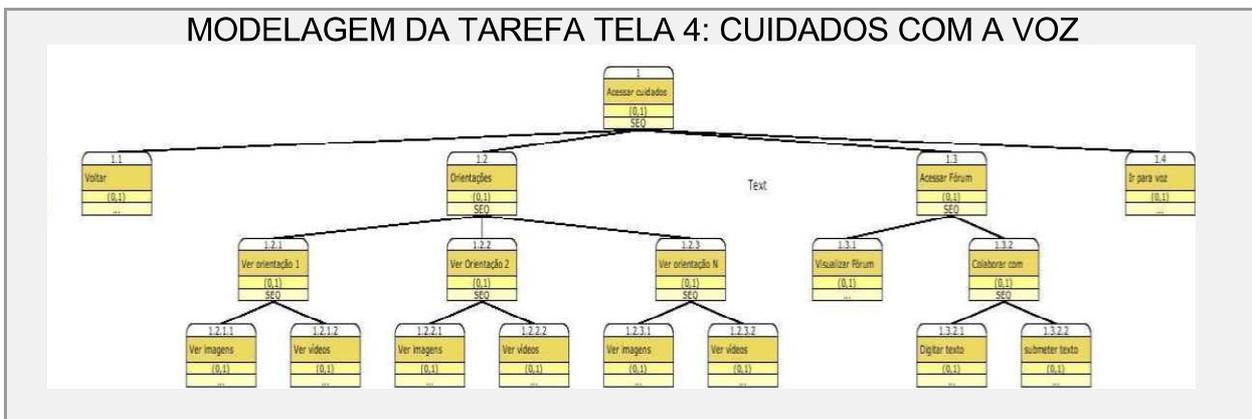


Figura 7: Modelagem da tela 4
Fonte: Elaborado pela autora (2017).

Na tela 4 é possível ter acesso ao conteúdo das orientações nos cuidados com a voz, acessar o fórum de discussões e submeter contribuição no fórum ou voltar para a tela anterior ou avançar para a próxima.

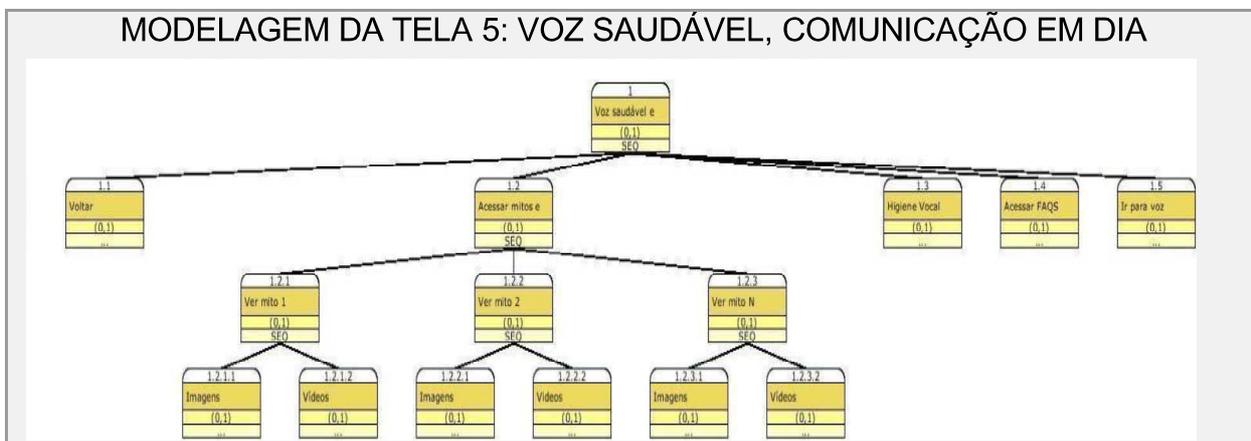


Figura 8: Modelagem da tela 5

Fonte: Elaborado pela autora (2017).

Na tela 5 encontra-se o acesso aos mitos e verdades nos cuidados com a voz, as orientações sobre higiene vocal e também é possível acessar as Faq's e retornar ou avançar para a próxima página.

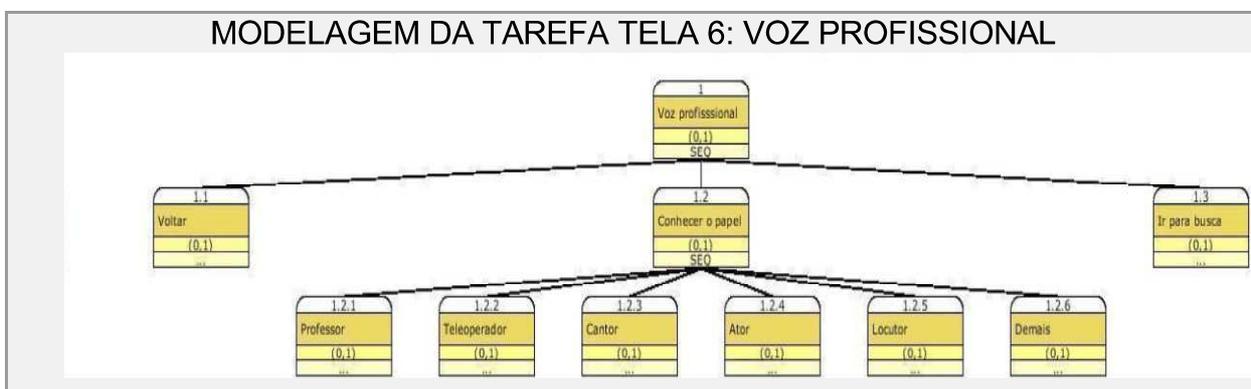


Figura 9: Modelagem da tela 6

Fonte: Elaborado pela autora (2017).

Na tela 6 encontram-se as informações referentes a voz profissional, com os diferentes papéis da voz nas profissões. Também é possível voltar ou avançar para a próxima tela.

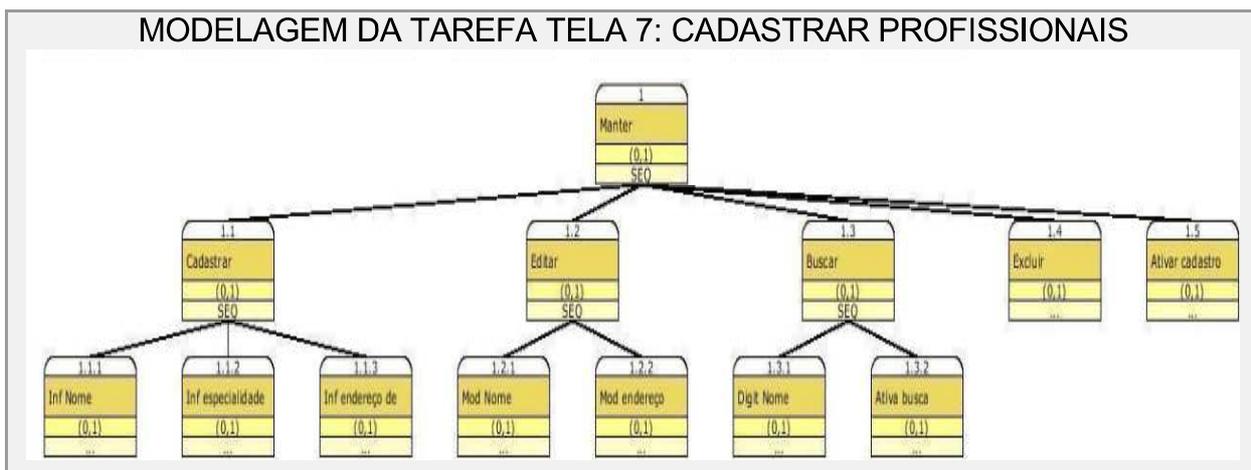


Figura 10: Modelagem da tela 7

Fonte: Elaborado pela autora (2017).

Na tela 7 encontram-se as informações sobre o cadastro de profissionais da área de voz, sendo possível manter profissionais, informar nome, especialidade e endereço e se o cadastro está ativo.

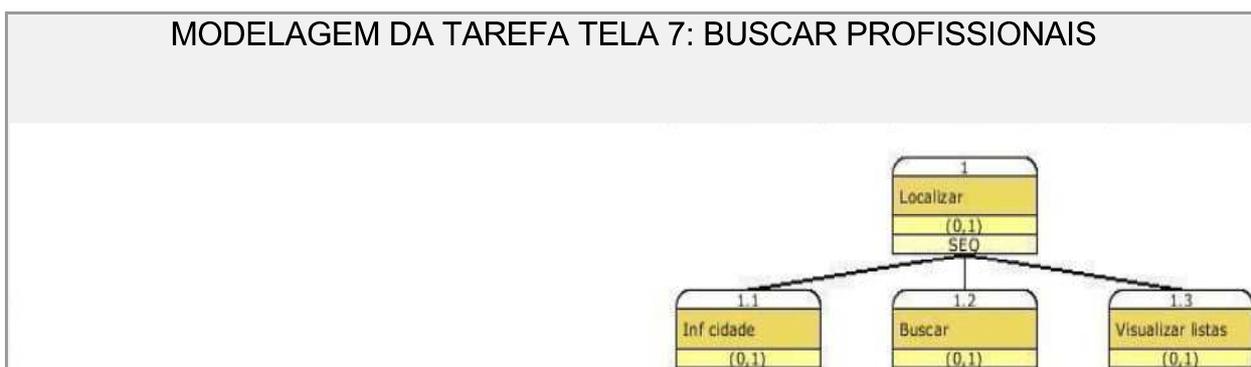


Figura 11: Modelagem da tela 7

Fonte: Elaborado pela autora (2017).

Nesta tela é possível fazer uma busca por profissionais na área desejada digitando o nome da cidade, aqui tem a opção voltar. Nesta etapa também são definidas as *UserStories* e os testes de aceitação, este artefato foi produzido pelo cliente, desenvolvedor e testador, mas as informações extraídas do usuário do *software* e do profissional da área também contribuíram. Este mecanismo segue no quadro 9.

| DEFINIÇÃO DAS USER STORIES E SEUS RESPECTIVOS TESTES DE ACEITAÇÃO | |
|---|---|
| US01 | Levantamento, verificação e validação inicial dos requisitos funcionais e não funcionais da aplicação. Para que assim se obtenham informações relativas às estimativas de tempo, financeira e de tecnologias a serem empregadas. |
| <p>TA1.1 – Levantar os requisitos funcionais e não funcionais</p> <p>TA1.2 - Buscar informações sobre YVOICE.</p> <p>TA1.3 – Avaliar as tecnologias que melhor se enquadram no desenvolvimento do projeto</p> <p>TA1.4 – Realizar exercícios usando a tecnologia estudada</p> | |
| Estimativa inicial: 8h | |
| US02 | Implementar o projeto visual da aplicação |
| <p>TA2.1 – Validar <i>wireframes</i> da aplicação</p> <p>TA2.2- Validar <i>wireframes</i> de forma estática em HTML</p> <p>TA2.3- Realizar a avaliação inicial de UX da aplicação preocupando-se com cores, fontes, contrastes.</p> <p>TA2.4- Finalizar roteamento estático das telas da aplicação.</p> | |
| Estimativa inicial: 2h | |
| US03 | Desenvolvimento da API do sistema |
| <p>TA3.1 – Definição dos dados que serão fornecidos pela API do sistema;</p> <p>TA3.1 – Criação dos <i>endpoints</i> a serem fornecidos pela API;</p> | |
| Estimativa inicial: 8h | |
| US04 | Implementação testes e validação das funcionalidades informações sobre anatomia do aparelho vocal no servidor |

TA4.1 – Buscar informações sobre anatomia do aparelho vocal (dados escritos, vídeos e imagens)

TA4.2.1 – Buscar palavra contida no glossário (a palavra pesquisada deve ser exibida)

TA4.2.2 Buscar palavra não existente no glossário (a palavra pesquisada não pode ser exibida)

Estimativa inicial: 8h

US05

Implementação testes e validação das funcionalidades informações sobre os inimigos da voz

TA5.1 – Buscar informações sobre inimigo número 1 (dados escritos, vídeos e imagens)

TA5.2- Buscar informações sobre inimigo número N (dados escritos, vídeos e imagens)

Estimativa inicial: 10h

US066

Implementação testes e validação das funcionalidades cuidados com a voz

TA6.1 –Buscar informações sobre orientações nos cuidados com a voz (dados escritos, vídeos e imagens)

TA6.2. - Conteúdo total do fórum de discussões

TA 6.2.1- Submeter contribuição válida para o fórum (atualizar lista de contribuições válidas do fórum)

TA6.2.2- Submeter contribuição inválida para o fórum (a contribuição não deve ser adicionada no fórum)

Estimativa inicial: 8h

US07

Implementação testes e validação das funcionalidades voz saudável, comunicação em dia

TA7.1 – Buscar informações sobre mitos e verdades sobre os cuidados com a voz dados escritos, vídeos e imagens)

TA7.2- Buscar informações sobre higiene vocal (dados escritos, vídeos e imagens)

TA7.3- Buscar informações sobre as FAQs

TA7.3.1 – Usuário ler e entender as FAQs.

Estimativa inicial: 6h

| | |
|---|--|
| US08 | Implementação testes e validação das funcionalidades voz profissional |
| TA8.1- Buscar informações sobre o papel da voz nas profissões (dados escritos, vídeos e imagens) | |
| Estimativa inicial: 8h | |
| US09 | Implementação testes e validação das funcionalidades manter profissionais |
| TA9.1- Cadastrar um profissional informando todos os dados (cadastro efetuado com sucesso) | |
| TA9.2- Cadastrar um profissional sem informar todos os dados (cadastro não efetuado) | |
| TA9.3- Fazer alteração no endereço de trabalho (mudança) | |
| TA9.4- Fazer busca pelo endereço de trabalho (endereço deve ser exibido) | |
| TA9.5- Fazer alteração na especialidade do profissional (cursos extras, aperfeiçoamento, especialização...) | |
| TA9.6- Excluir cadastro do profissional (por motivo de morte ou outro) | |
| Estimativa Inicial: 4h | |
| US010 | Implementação testes e validação das funcionalidades guia simples de ajuda |
| TA10.1- Procurar informações diversas no guia de ajuda | |
| Estimativa inicial: 8h | |
| US11 | Prototipação das telas desenvolvidas estaticamente no IONIC |
| TA11- Criação do ambiente desenvolvimento em IONIC | |
| TA11.1- Prototipação das telas estáticas em componentes Angular a serem gerenciados pelo IONIC | |
| TA12.2- Fluxo de telas no sistema utilizando <i>mocks</i> em para fornecimento dos dados | |
| Estimativa Inicial: 4h | |
| US012 | Integração entre <i>frontend</i> e <i>backend</i> da funcionalidade de informações sobre anatomia do aparelho vocal |

TA12 – Buscar informações sobre anatomia do aparelho vocal (dados escritos, vídeos e imagens) deve

TA12.1 – Buscar palavra contida no glossário (a palavra pesquisada deve ser exibida)

TA12.1.2 Buscar palavra não existente no glossário (a palavra pesquisada não pode ser exibida)

Estimativa inicial: 8h

US013

Integração entre *frontend* e *backend* da funcionalidade informações sobre os inimigos da voz

TA13 – Buscar informações sobre inimigo número 1 (dados escritos, vídeos e imagens)

TA13.1- Buscar informações sobre inimigo número N (dados escritos, vídeos e imagens)

Estimativa Inicial: 4h

US014

Integração entre *frontend* e *backend* da funcionalidade informações sobre os cuidados com a voz

TA14 – Requisitar informações sobre orientações nos cuidados com a voz (dados escritos, vídeos e imagens) via API e expor no *frontend*;

TA14.1 Requisitar via API o conteúdo total do fórum de discussões e expor no *frontend*

TA 14.2- Submeter contribuição válida para o fórum (atualizar lista de contribuições válidas do fórum)

TA14.2.1- Submeter contribuição inválida para o fórum (a contribuição não deve ser adicionada no fórum).

Estimativa Inicial: 4h

US015

Integração entre *frontend* e *backend* da funcionalidade informações sobre voz saudável, comunicação em dia

TA15 – Buscar informações sobre mitos e verdades sobre os cuidados com a voz dados escritos, vídeos e imagens)

TA15.1- Buscar informações sobre higiene vocal (dados escritos, vídeos e imagens)

TA15.2- Buscar informações sobre as FAQs

TA15.2.1 – Usuário ler e entender as FAQs

Estimativa Inicial: 4h

US016 | **Integração entre *frontend* e *backend* da funcionalidade informações sobre voz profissional**

TA16- Buscar informações sobre o papel da voz nas profissões (dados escritos, vídeos e imagens)

Estimativa Inicial: 2h

US017 | **Integração entre *frontende backend* da funcionalidade manter profissionais**

TA17- Cadastrar um profissional informando todos os dados (cadastro efetuado com sucesso)

TA17.1- Cadastrar um profissional sem informar todos os dados (cadastro não efetuado)

TA17.2- Fazer alteração no endereço de trabalho (mudança)

TA17.3- Fazer busca pelo endereço de trabalho (endereço deve ser exibido)

TA17.4- Fazer alteração na especialidade do profissional (cursos extras, aperfeiçoamento, especialização...)

TA17.5- Excluir cadastro do profissional (por motivo de morte ou outro)

Estimativa Inicial: 10h

US018 | **Integração entre *frontend* e *backend* da funcionalidade guia simples de ajuda.**

TA18- Procurar informações diversas no guia de ajuda

Estimativa Inicial: 7h

US019 | **Validação das integrações realizadas**

TA19 – Testes de aceitação

TA19.1 – Avaliação final da Usabilidade do sistema

TA19.2 - Versão Beta teste gerada palavra pesquisada não pode ser exibida)

Estimativa Inicial: 7h

Quadro 9: *UserStories* e Testes de Aceitação

Fonte: Elaborado pela autora (2017).

. O protótipo de interface é um elemento de grande importância para a usabilidade do *software*, é uma representação da interface com a qual o usuário pode compartilhar e traz os dados necessários para indicar mudanças e progressos, pois a interface com o usuário é uma condição indispensável para o sucesso do *software*, a fabricação dos protótipos trouxe a contribuição do cliente, aluno, equipe multidisciplinar, professor mediador e desenvolvedor. Este mecanismo foi determinado através dos métodos do design utilizando a ferramenta *Pencil*, e está especificado nas figuras 12, 13, 14, 15, 16, 17 E 18.



Figura 12: Tela Inicial
Fonte: Elaborado pela autora (2017).



Figura 13: Tela 2 Cuidados com a Voz
Fonte:Elaborado pela autora (2017).



Figura 14: Tela 3 Inimigos da Voz
Fonte: Elaborado pela autora (2017).



Figura 15: Tela 4 Cuidados com a Voz
Fonte: Elaborado pela autora (2017).



Figura 16: Tela 5 Voz Saudável, Comunicação em dia
Fonte: Elaborado pela autora (2017).



Figura 17: Tela 6 Voz Profissional
Fonte: Elaborado pela autora (2017).



Figura 18: Tela 7 Buscar Profissionais
Fonte: Elaborado pela autora (2017).

O Projeto arquitetural traz a estrutura dos objetos de dados e programas que são essenciais para arquitetar um sistema baseado em computador. Ele descreve o caráter arquitetural que o sistema seguirá, a estrutura e os atributos dos componentes que compõem o sistema. É construído a partir da conversa com o cliente e os componentes que foram desenvolvidos anteriormente. O projeto arquitetural do YVOICE está sendo

apontado no quadro 10 e foi realizado com base no questionário do professor Jacques Sauvêdo Departamento Sistemas e Computação da UFCG - Perguntas a Fazer ao Elaborar um Projeto Arquitetural.

| SOBRE ENTIDADES EXTERNAS AO SISTEMA |
|--|
| <p>Quais sistemas externos devem ser acessados? De início API responsável pela persistência de dados do próprio sistema. Além das APIs de servidores de vídeo como Youtube e Vimeo</p> <p>Como serão acessados? A persistência no SGBD será auxiliada pelo MySQL, já o acesso aos dados será feito via REST</p> <p>Há integração com o legado a ser feita? Como? Não</p> |
| DETERMINAÇÃO DE OPORTUNIDADES PARA O REUSO DE <i>SOFTWARE</i> |
| <p>Há interesse/conveniência/tempo em aproveitar tais oportunidades? SIM!</p> <p>Como isso pode ser feito? Sim, de início com os componentes padrão das bibliotecas de desenvolvimento mobile. Além disso, o <i>backend</i> da aplicação será desenvolvido seguindo o modelo REST, desta forma, as regras de negócio tanto para a versão iOS quanto Android serão a mesma disponibilizadas via API.</p> <p>Construindo um framework? Não.</p> |

Quadro 10: ProjetoArquitetural

Fonte:Elaborado pela autora (2017).

| SOBRE A ORGANIZAÇÃO GERAL DO SISTEMA |
|---|
| <p>O sistema é centralizado ou distribuído? Distribuído</p> <p>Como modularizar em subsistemas? Será dividido em uma camada de persistência, outra será o REST, além da camada de <i>frontend</i> desenvolvida em IONIC, unificando assim as versões Android e iOS uma aplicação híbrida.</p> <p>Minimizar acoplamento entre os pedaços? Nas camadas de <i>backend</i> será utilizado o princípio SOLID diminuindo assim o acoplamento e dando mais flexibilidade às classes. Já no front, será feito uso do AngularJS, trabalhando assim com o conceito de componentes web.</p> <p>Tais camadas serão lógicas ou terão existência física separada?</p> <p>Camada Gráfica: corresponde à interface com o usuário; são as telas do <i>software</i> que permitirão uma boa usabilidade por parte do usuário do app.</p> <p>Camada de Negócio: é nela que se encontra o algoritmo do <i>software</i>. É também essa camada que permitirá uma comunicação com a interface através de <i>endpoints</i> seguindo o princípio REST.</p> <p>Camada de Persistência de Dados: A persistência será feita em via SQL através da chamada de servidor.</p> <p>Qual é o nível de acoplamento, frequência de interações, volumes de dados trocados entre as camadas? Baixo/ razoável/ poucos</p> <p>A programação será feita com qual paradigma? OO? Sim!</p> <p>Que linguagens e ferramentas serão usadas? Na persistência será feito uso de SQL, onde a aplicação do servidor será desenvolvida em NodeJS. Já no <i>frontend</i> será feito uso de AngularJS para desenvolvimento dos componentes e interfaces tendo o IONIC como container da aplicação.</p> <p>Como será feito a alocação de memória para os elementos do programa? Estaticamente, com exceção dos dados persistidos através do SGBD.</p> <p>Há considerações especiais de segurança que afetam o sistema? Não</p> |
| SOBRE A CAMADA DE INTERFACE |

O sistema será acessado usando que tipos de clientes? Público adulto.
Como fazer a interface gráfica? Via desenvolvimento de uma aplicação híbrida, usa-se os componentes AngularJS tendo o Ionic como suporte à aplicação.
Com que ferramentas? VS Code
Com que componentes visuais? Serão usadas imagens e vídeos.
Serão desenvolvidos? Comprados? Todas as imagens e vídeos serão criados pela equipe de desenvolvimento e/ou baixadas da internet – desde que sejam liberadas para quaisquer propósitos pelos autores.
Há considerações de localização (para outras línguas ou outros países)? Não.

SOBRE A CAMADA DE LÓGICA DA APLICAÇÃO

Quais são os componentes principais a fazer? Camada de Negócio, pois nesta camada se encontra o algoritmo do *software*. É também essa camada que permitirá uma comunicação com a interface (através de uma classe que servirá de ponte) e os dados.

Há uso de *threads*? Sim, contudo as mesmas serão caracterizadas por requisições assíncronas à API de *backend*.

Quadro 11: Projeto Arquitetural (Continuação)

Fonte: Elaborado pela autora (2017).

SOBRE A CAMADA DE DADOS PERSISTENTES

Quais são as fontes de dados? Externas? Internas? Existentes? Novas? Imagens e áudio serão internos (algumas existentes e outras criadas pela equipe). Haverá informações de origem externa – dados do usuário para recordes.

Que estratégia de persistência será usada:

Em arquivos? Imagens e áudio, sim.

Usando um SGBD? Dados do usuário para recordes.

Qual paradigma de SGBD usar? Relacional.

Como será feita a população dos bancos de dados? Através da aplicação.

PERGUNTAS ADICIONAIS

Como será resolvida a instalação do produto? O programa ser portátil. Um *.apk que deverá ser gerado de acordo com o sistema operacional do aparelho móvel

O que será comprado, feito, modificado? Nada/classes e imagens/algumas imagens.

Como será a distribuição do *software*? Via lojas virtuais para Apps

Quadro 12: Perguntas a Fazer ao Elaborar um Projeto Arquitetural

Fonte: Elaborado pela autora (2017).



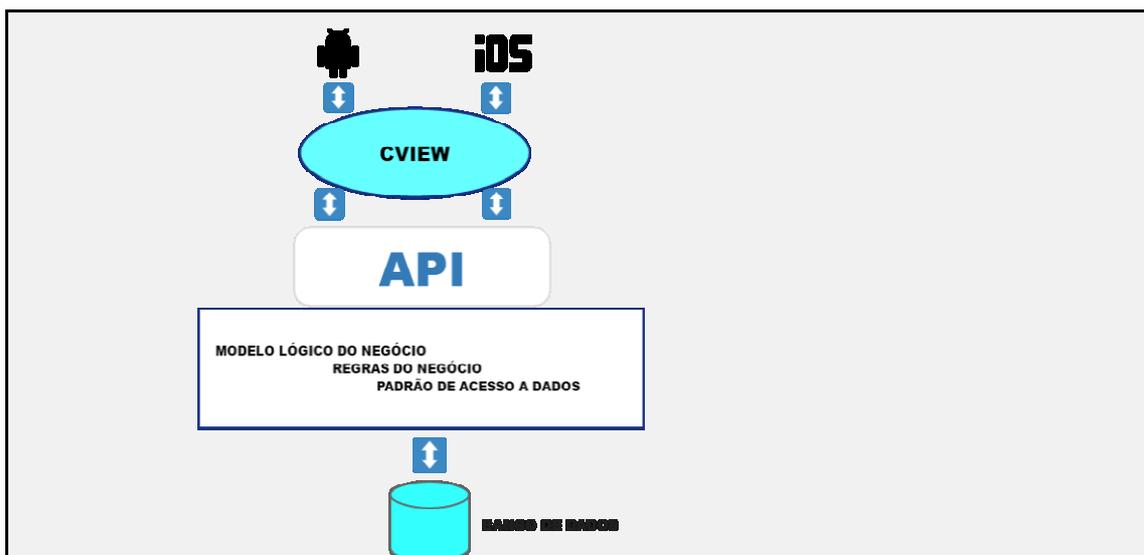


Figura 19: Perguntas a Fazer ao Elaborar um Projeto Arquitetural

Fonte (Questionário):Sauvé, Jacques - Perguntas a Fazer ao Elaborar um Projeto Arquitetural.

A Modelagem dos dados baseia-se em desenhar o sistema de informações, sistematizando as entidades lógicas e as dependências lógicas entre essas entidades, deve ter autonomia no uso da tecnologia implementada por causa da frequente mudança dos produtos tecnológicos. A produção deste artefato foi responsabilidade do desenvolvedor. O modelo lógico de dados do YVOICE está representado na figura 19.

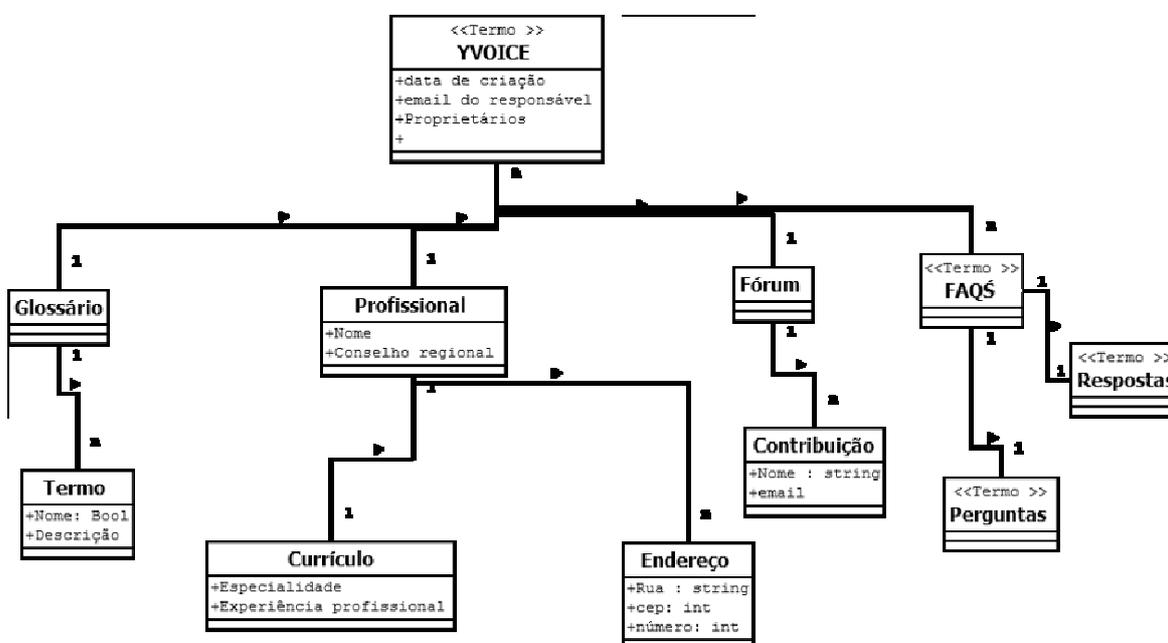


Figura 20: Modelo Lógico de Dados

Fonte: Elaborado pela autora (2017).

Quando são finalizadas as atividades de inicialização, inicia-se o planejamento do projeto. Onde toda a equipe do projeto participa do planejamento do release e o da

interação, tais componentes estão sendo mostrados nos quadros 12 e 13 respectivamente.

| | | |
|----------------------------|----------------------------------|----------------|
| RELEASE 01: 10 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 01 | USO 01 | |
| Iteração 02 | USO 02 | |
| RELEASE 02: 13 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 03 | USO 03 | |
| Iteração 04 | USO 04 | |
| RELEASE03: 17 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 05 | USO 05 | |
| Iteração 06 | USO 06 | |
| RELEASE 04: 12 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 07 | USO 07 | |
| Iteração 08 | USO 08 | |
| Iteração 09 | USO 09 | |
| RELEASE 05: 14 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 10 | USO 10 | |
| RELEASE 06: 19 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 11 | USO 11 | |
| RELEASE 07: 8 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 12 | USO 12 | |
| RELEASE 08: 20 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 13 | USO 13 | |
| RELEASE 09: 7dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 14 | USO 14 | |
| RELEASE 10: 7dias | GERENTE: TARCIANA PALETOT | |

| Iteração | <i>UserStory</i> | Período |
|----------------------------|----------------------------------|----------------|
| Iteração 15 | USO 15 | |
| RELEASE 11: 9 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 16 | USO 16 | |
| RELEASE 12: 10 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 17 | USO 17 | |
| RELEASE 13: 5 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 18 | USO 18 | |
| RELEASE 14: 7 dias | GERENTE: TARCIANA PALETOT | |
| Iteração | <i>UserStory</i> | Período |
| Iteração 19 | USO 19 | |

Quadro 13: Plano de *Release*

Fonte: Elaborado pela autora (2017).

PLANO DE ITERAÇÃO

Um conjunto de atividades e tarefas divididas por sequências de tempo, com recursos atribuídos e dependências de tarefas, para a iteração; é um plano sofisticado e bem elaborado que planeja e ajuda a equipe de desenvolvimento nos objetivos a serem alcançados em cada iteração.

INTERAÇÃO 01

USO 01 – Levantamento, verificação e validação inicial dos requisitos funcionais e não funcionais da aplicação. Para que assim se obtenham informações relativas às estimativas de tempo, financeira e de tecnologias a serem empregadas.

| TESTES DE ACEITAÇÃO | | STATUS |
|----------------------------|--|---------------|
| TA1.1 | Levantar os requisitos funcionais e não funcionais | |
| TA1.2 | Buscar informações sobre YVOICE. | |
| TA1.3 | Avaliar as tecnologias que melhor se enquadram no desenvolvimento do projeto | |
| TA1.4 | Realizar exercícios usando a tecnologia estudada | |

| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
|-------------|--|-------------|---------------------|------------|--------|
| A1.1 | Buscar material sobre a tecnologia estudada | Tarciana | | | |
| A1.2 | Realizar exemplos sobre as tecnologias estudadas | Tarciana | | | |
| A1.3 | Validar junto ao cliente as escolhas tecnológicas feitas | Tarciana | | | |
| A1.4 | Pesquisar e produzir literatura sobre dados escritos e biblioteca para vídeos e imagens sobre anatomia do aparelho vocal | Tarciana | | | |
| A1.5 | Pesquisar e produzir literatura sobre dados escritos e biblioteca para vídeos e imagens sobre os inimigos da voz | Tarciana | | | |
| A1.6 | Pesquisar e produzir literatura sobre dados escritos e biblioteca para vídeos e imagens sobre os cuidados com a voz | Tarciana | | | |
| A1.7 | Pesquisar e produzir literatura sobre dados escritos e biblioteca para | Tarciana | | | |

| | | | | | |
|-------------|--|----------|--|--|--|
| | vídeos e imagens sobre voz saudável, comunicação em dia | | | | |
| A1.8 | Pesquisar e produzir literatura sobre dados escritos e biblioteca para vídeos e imagens sobre voz profissional | Tarciana | | | |
| A1.9 | Pesquisar e produzir literatura sobre dados escritos e biblioteca para vídeos e imagens sobre buscar profissionais | Tarciana | | | |

Quadro 14: Plano de Interação

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 02

USO 02 -Implementar o projeto visual da aplicação

| TESTES DE ACEITAÇÃO | | <i>STATUS</i> |
|---------------------|---|---------------|
| TA2.1 | Avaliar os <i>wireframes</i> da aplicação | |
| TA2.2 | Prototipação do <i>wireframes</i> de forma estática em HTML | |
| TA2.3 | Realizar a avaliação inicial de UX da aplicação preocupando-se com cores, fontes, contrastes. | |

| TA2.4 | | Verificar estático das telas de aplicação | | | |
|-----------|--|---|---------------------|------------|--------|
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A2.1 | Implementar a interface para a tela inicial do app | Tarciana | | | |
| A2.2 | Implementar a interface para a tela informações sobre anatomia do aparelho vocal . | Tarciana | | | |
| A2.3 | Implementar a interface para a tela informações sobre inimigos da voz . | Tarciana | | | |
| A2.4 | Implementar a interface para a tela informações sobre cuidados com a voz | Tarciana | | | |
| A2.5 | Implementar a interface para a tela informações sobre voz saudável, comunicação em dia | Tarciana | | | |
| A2.6 | Implementar a interface para a tela informações sobre voz profissional | Tarciana | | | |
| A2.7 | Implementar a | Tarciana | | | |

| | | | | | |
|------|--|----------|--|--|--|
| | interface para a tela informações sobre buscar profissionais | | | | |
| A2.8 | Finalizar roteamento estático das telas de aplicação | Tarciana | | | |

Quadro 15: Implementação do Projeto.

Fonte: Elaborado pela autora (2017).

Os demais planos de iteração com o detalhamento de seus artefatos para as iterações de 3 a 19 se encontram em anexo.

O teste de Usabilidade examina a facilidade do *software* de ser compreendido e também analisa se as especificidades dos clientes e do usuário foram atendidas. Tal artefato segue no quadro 16.

| ROTEIRO DAS ATIVIDADES | |
|----------------------------------|--|
| DESCRIZAÇÃO INICIAL | <p>Neste Teste de Usabilidade, você assumirá as funções de usuário do aplicativo. Este <i>software</i> traz uma proposta de dicas para os cuidados com a voz.</p> <p>OBS.: Caso encontre alguma dificuldade que não comprometa a realização da atividade, não se preocupe e siga em frente.</p> |
| ATIVIDADE 01: Abrir tela inicial | <p>Roteiro: Nesta atividade você irá entrar no YVOICE através da tela inicial.</p> <p>Instruções</p> <ul style="list-style-type: none"> ● Abra o <i>software</i> clicando duas vezes no ícone referente ao mesmo; |

| | |
|---|---|
| | <ul style="list-style-type: none"> ● Verifique se o programa abriu corretamente; ● Observe a tela inicial; ● Avance para a próxima tela; |
| <p>ATIVIDADE 02: Analisar tela anatomia do aparelho vocal</p> | <p>Roteiro: Nesta atividade você irá testar as funcionalidades da tela anatomia do aparelho vocal</p> <p>Instruções:</p> <ul style="list-style-type: none"> ● Observe se está em anatomia... ● Acessar guia aparelho fonador <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens ● Acessar Glossário ● Pesquisar palavra no glossário <ul style="list-style-type: none"> ➤ Verificar resultado da busca ● Voltar para a tela anterior ● Avançar para a próxima tela |
| <p>ATIVIDADE 03: Analisar tela inimigos da voz</p> | <p>Roteiro: Nesta atividade você irá testar as funcionalidades da tela anatomia do aparelho vocal</p> <p>Instruções:</p> <ul style="list-style-type: none"> ● Observe se está em inimigos... ● Acessar guia inimigo 1 <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens ● Acessar guia inimigo N <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens ● Voltar para a tela anterior ● Avançar para a próxima tela |

| | |
|---|--|
| <p>ATIVIDADE 04: Analisar tela Cuidados com a voz</p> | <p>Roteiro: Nesta atividade você irá testar as funcionalidades da tela Cuidados com a voz</p> <p>Instruções</p> <ul style="list-style-type: none"> ● Observe se está em cuidados... ● Acessar guia orientações <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens ● Acessar guia fórum de discussões <ul style="list-style-type: none"> ➤ Ler tópicos do fórum ➤ Submeter contribuição no fórum ● Voltar para a tela anterior ● Avançar para a próxima tela |
| <p>ATIVIDADE 05: Analisar tela voz saudável, comunicação em dia</p> | <p>Roteiro: Nesta atividade você irá testar as funcionalidades da tela voz saudável</p> <p>Instruções:</p> <ul style="list-style-type: none"> ● Observe se está em voz saudável ● Acessar guia mitos e verdades <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens ● Acessar guia orientações sobre higiene vocal <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens ● Acessar guia FAQ'S <ul style="list-style-type: none"> ➤ Ler perguntas e respostas ● Voltar para a tela anterior ● Avançar para a próxima tela |
| <p>ATIVIDADE 06: Analisar tela voz profissional</p> | <p>Roteiro: Nesta atividade você irá testar as funcionalidades da tela voz profissional</p> <p>Instruções:</p> <ul style="list-style-type: none"> ● Observe se está em voz profissional |

| | |
|---|---|
| | <ul style="list-style-type: none"> • Acessar guia voz profissional <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens • Voltar para a tela anterior • Avançar para a próxima tela |
| <p>ATIVIDADE 06: Analisar tela buscar profissionais</p> | <p>Roteiro: Nesta atividade você irá testar as funcionalidades da tela buscar profissionais</p> <p>Instruções:</p> <ul style="list-style-type: none"> • Observe se está em voz profissional • Acessar guia voz profissional <ul style="list-style-type: none"> ➤ Ler texto ➤ Assistir vídeos ➤ Visualizar imagens • Voltar para a tela anterior <ul style="list-style-type: none"> ➤ Avançar para a próxima tela |

Quadro 16: Teste de Usabilidade

Fonte: Elaborado pela autora (2017).

4 CONCLUSÕES E CONSIDERAÇÕES FINAIS

4.1 CONTRIBUIÇÕES

O desenvolvimento descrito no presente estudo possibilitou a proposição de uma arquitetura de um aplicativo que permita, de maneira dinâmica e instrucional, auxiliar os professores nos cuidados relacionados a voz. Nesse sentido, o passo inicial foi a realização de uma revisão da literatura visando suportar os conceitos inerentes a proposta do trabalho. Em seguida, alicerçando a proposição da arquitetura, encontra-se a adoção do processo de *softwareYPeduc*: um modelo de processo que procura simplificar o desenvolvimento de um *software* educativo, tendo como base as metodologias ágeis que consideram que uma aplicação está passiva de mudanças e adaptações aos diversos cenários.

A proposição da arquitetura, mesmo que em um estágio embrionário, induziu a construção de protótipos que atenderam às expectativas da modelagem dos requisitos, gerando, de certo modo, resultados satisfatórios e permitindo que o aplicativo naturalmente entre em produção, devendo ser implementados e melhorados no futuro, por uma equipe multidisciplinar, uma vez que a arquitetura foi idealizada considerando flexibilidade, podendo ser expandida e melhorada.

Ao longo deste trabalho surgiram novas possibilidades que não foram desenvolvidas, pois tornariam este trabalho ainda mais extenso. A ausência de uma equipe impossibilita o avanço do mesmo em um ritmo sustentável de entregas incrementais. Nessa perspectiva, o desenvolvimento do presente estudo sinalizou como a utilização das ferramentas adequadas permite a realização de um trabalho árduo, solo, de forma mais eficiente, diminuindo o tempo de produção dos artefatos. É indispensável citar o fato de que além dos aspectos inerentes da engenharia de um *software*, aspectos da engenharia da aprendizagem definitivamente devem ser levados em conta, de modo que contribuam para a confecção de um programa de forma a torná-lo acessível a todos.

Pretende-se, portanto, que este trabalho sirva de contribuição para reflexões que possam tirar proveito dos conhecimentos e dos autores aqui referenciados. Por fim, espera-se que o mesmo possa criar lacunas e suscitar questionamentos que induzam a incursões no sentido de se chegar a um maior entendimento sobre novas formas de utilização das tecnologias digitais a serviço da educação, com aplicabilidade à saúde, o que corrobora com a multidisciplinaridade inerente ao novo paradigma em que se coloca a humanidade.

4.2 LIMITAÇÕES

Por se tratar de uma monografia, o pesquisador assumiu praticamente todos os papéis do projeto, tendo o estudo de caso uma proporção considerada pequena mediante o que realmente explicita um desenvolvimento real de um *software*. Vale a pena reforçar que o intuito foi não apenas verificar se todos os pontos importantes foram acomodados no processo, mas, primordialmente, definir de modo detalhado a arquitetura, deixando a desejar na implementação prevista na metodologia.

Ainda sobre o estudo de caso, não houve possibilidade de interagir com uma equipe multidisciplinar proposta para fundamentar os requisitos de um *software* educacional voltado para a saúde vocal, havendo assim, uma perda no processo de

concepção do *software* no tocante às atividades de levantamento de requisitos. O tempo foi, sem dúvida, outro fator limitante no desenvolvimento deste trabalho, não sendo possível, ainda, iniciar a implementação das especificações desejadas para o *software* YVOICE, mesmo com a sua arquitetura bem definida e com o planejamento das *releases* e das iterações já estabelecido.

4.3 TRABALHOS FUTUROS

As contribuições apresentadas já representam um certo grau de avanço dentre os mais variados usos de *software* educacional, neste caso específico, voltado para a saúde vocal. Apesar dos benefícios aqui já apresentados, por meio da experiência de projeto vivenciada, e da proposição da arquitetura, este trabalho deve ser continuado através do desenvolvimento do *software* YVOICE, em um uso natural do processo YPeduc. Com o prosseguimento de tais estudos podem surgir novas estratégias de refinamento dos critérios essenciais elencados neste trabalho, uma vez que existem poucos *softwares* voltados para os cuidados com a voz. O uso desta metodologia neste tipo de trabalho se torna inédito, não somente por tais artefatos, mas também pela conotação educacional da ferramenta YVOICE, com ênfase na consideração de aspectos voltados à saúde.

É importante ressaltar também que o YVOICE seja avaliado na sua totalidade, tanto em aspectos relacionados à qualidade de código e ao uso de padrões, quanto de aspectos relacionados à usabilidade, à comunicabilidade e, pelo fato de ser um *software* educativo, de aspectos relacionados à aprendizagem.

REFERÊNCIAS

AGILE MANIFESTO. Manifesto for Agile Software Development. **Agile Alliance**: 2001. Disponível em: <<http://www.agilemanifesto.org/>>. Acesso em: 11 ago. 2017.

ANDREATTA-DA-COSTA, L.; RAMIRO, F.S. O Estudo de Estruturas Hiperestáticas à Luz da Teoria da Aprendizagem Significativa. **Revista Liberato**, Novo Hamburgo, v. 18, n. 29, jan./jun., 2017, p. 01-132.

BARRETO, J. M. **Inteligência Artificial no Limiar do Século XXI**. 2.ed. Florianópolis: Duplic, 1999. 296p.

BEHLAU, M. **Voz: o livro do especialista**. 1 ed. Revinter, 2001. 348 p.

BEHLAU, M.; PONTES, P. **Avaliação e Tratamento das Disfonias**. São Paulo: Lovise, 1995. 312 p.

BEHLAU, M.; MADAZIO, G.; FEIJÓ, D.; AZEVEDO, R.; GIELOW, I.; REHDER, M.I. Aperfeiçoamento Vocal e Tratamento Fonoaudiológico das Disfonias. In: Behlau, M. (org). **Voz: o livro do especialista**. Rio de Janeiro: **Revinter**, 2005. p. 409-437.

BEHLAU, M.; NAGANO L.; DRAGONE, M.L.S. **A Voz que Ensina: o professor e a comunicação oral em sala de aula**. Rio de Janeiro: Revinter, 2004. 68 p.

BRAINIAC; DUONG THÁNH AN. **Pencil Project 3.0.4**. Evolus: CDRoom. O Pencil é uma ferramenta de prototipagem de GUI livre e de código aberto, possível de instalar e usar facilmente para maquetes em plataformas desktop populares. 2017.

BRASIL. MINISTÉRIO DA SAÚDE, SECRETARIA EXECUTIVA. **Departamento de Informática do Sus – DATASUS Trajetória 1991-2002**. Brasília: Ministério da Saúde, 2002. (Série G. Estatística e Informação em Saúde).

CARDOSO, J., P.; ROSA, V. A.; LOPES, C.R.S.; VILELA, A.B.A. SANTANA, A. S.; SILVA, S.T. da. Construção de Uma Práxis Educativa em Informática na Saúde para Ensino de Graduação. **Ciência e Saúde Coletiva** [online], v.13, n.1, Rio de Janeiro, jan./fev., 2008. p. 283-288.

CHOI-CARDIM, K.; BEHLAU, M.; ZAMBON, F. Sintomas Vocais e Perfil de Professores em um Programa de Saúde Vocal. **Revista CEFAC** [online]. vol.12, n.5, p.811-819, abr., 2010.

CONSTANTINO, T.; Guimarães, I. Influência da Duração da Disfonia na Qualidade Vocal e seu Impacto Psicossocial em Mulheres. Re(habilitar). **Revista da ESSA**, n. 1, Edições Colibri, p. 3-24. 2005.

DEPARTAMENTO DE ENGENHARIA ELÉTRICA, UNIVERSIDADE FEDERAL DE CAMPINA GRANDE. Formulário para o levantamento do Perfil do Usuário, 2003, Laboratório de Interfaces Homem-Máquina. **Departamento de Engenharia Elétrica**: 2003. Disponível em: http://www.dee.ufcg.edu.br/~scaico/facisa/ihm/perfil_do_usuario.pdf; Acesso em: 11 ago. 2017.

DEPARTAMENTO DE INFORMÁTICA DO SUS – DATASUS. Histórico e Apresentação. **Portal da Saúde – SUS**: 2017. Disponível em: <<http://datasus.saude.gov.br/datasus>>. Acesso em: 11 ago. 2017.

GAMEZ, L. **TICESE - Técnica de inspeção de conformidade ergonômica de software educacional**. 1998. Dissertação (Mestrado), Universidade do Minho. Portugal, 1998.

KOOIJMAN, P.G.C.; DE JONG, F.I.; C.R.S.; THOMAS, G.; HUINCK, W.; DONDEERS, R.; GRAAMANS, K.; SCHUTTE, H.K. Risk factors for voice problems in teachers. **Folia PhoniatrLogop**, 2006, n. 58: p. 159-174.

MARCON, J. P. F.; DIAS, T. P. DEEPWEB: O Lado Sombrio da Internet. **Conjuntura Global**, v. 3, n. 4, out./dez., 2014, p. 233-243.

MARÇAL, C. C. B.; PERES, M. A. Alteração Vocal auto-referida em Professores: prevalência e fatores associados. **Revista de Saúde Pública**, São Paulo, v. 45, n. 3, p. 503-511, abr., 2011.

MEDEIROS, A.F. **Elicitação de Requisitos Essenciais para uma Metodologia Ágil de Desenvolvimento de Software Educativo**. 2012. 83 f. Universidade Estadual da Paraíba, 2012.

MENDES, A. P.; FERREIRA, L. J.L.; CASTRO, E. Softwares e Hardwares de Análise Acústica da Voz e da fala. **Distúrbios da Comunicação**, São Paulo, v. 24, n. 3, dez., 2012, p. 421-430.

PROGRAMA DE EDUCAÇÃO TUTORIAL, UNIVERSIDADE FEDERAL DE CAMPINA GRANDE. **easYProcess: Um processo de desenvolvimento de software**.

Apostila. 2007. Disponível em: < <http://www.mps.xpg.com.br/YP.pdf> >. Acesso em: 14 de abril de 2016.

PETRÔNIO, F.; LULA JÚNIOR, B.; MEDEIROS, A.; SILVA JÚNIOR, N.J.O. **iTAOS** - Ferramenta para Modelagem de Tarefas. 2010. In: Simpósio de Fatores Humanos em Sistemas Computacionais, 2010. **Anais...** Simpósio de Fatores Humanos em Sistemas Computacionais.

PINHO, S. M. R. **Fundamentos em Fonoaudiologia: tratando os distúrbios da voz.2.** ed. Rio de Janeiro: Guanabara Koogan; 2003. 150 p.

PINOCHET, L.H.C. Tendências de Tecnologia de Informação na Gestão da Saúde. **O Mundo da Saúde**, São Paulo: v. 35, n. 4, 2011, p.382-394.

PRESSMAN, R. S. **Engenharia de Software.**7 ed. São Paulo: Makron Books, 1995. 1056 p.

_____. **Software Engineering: a practitioner's approach.** 5th ed. Boston: McGraw-Hill Higher Educacion, 2001. 860 p.

ROCHA, A. R.C. (org.). **Qualidade de Software: teoria e prática.** São Paulo: Prentice Hall, 2001. 3003 p.

SABBATINI R.M. O Ensino da Informática Aplicada à Medicina: roteiros e bibliografia básica. **Revista Informédica.** v. 2, n. 8, 1994, p. 5-12.

SAUVÉ, J. Perguntas a Fazer ao Elaborar um Projeto Arquitetural. **UFCCG:** 2012. Disponível em: <http://jacques.dsc.ufcg.edu.br/cursos/proj/gerenciadesenv/arquitetural.htm>. Acesso em: 11 ago. 2017.

SERVILHA E.A.M. Voz na infância. In: FERREIRA L.P.; LOPES, D.M.B.; LIMONGI, S.C.O. (org.). **Tratado de fonoaudiologia.** São Paulo: Roca; 2004. p.122-6.

SILVA, J.M. Novas Tecnologias em Sala de Aula. **Revista Ciência, Salud, Educación y Economía,** nº 10, 2016.

SOFFA, M.M.; ALCÂNTARA, P.R.C. O Uso do Software Educativo: reflexões da prática docente na sala informatizada. In: Congresso Nacional de Educação (EDUCERE), 8., 2008. Curitiba. **Anais eletrônicos...** Curitiba: PUCPR, 2008. Disponível em:

<http://www.pucpr.br/eventos/educere/educere2008/anais/pdf/335_357.pdf>. Acesso em: 10 maio 2015.

SOMMERVILLE, I. **Engenharia de Software**. 8. ed. São Paulo: Pearson, 2007. 568 p.
VALENTE, A.J. **Análise dos Diferentes Tipos de Softwares Usados na Educação**. 1 ed. Campinas: Nied/Unicamp, 1999, p. 89-110.

VALENTE, A.J. (org.). O Computador na Sociedade do Conhecimento. Campinas: UNICAMP/NIES, 1999. 85 p.

VIERA, F.M.S. Avaliação de Software Educativo: reflexões para uma análise de critérios. **Ignição Digital**: 1999. Disponível em:
<<http://www.edutecnet.com.br/edmagali2.htm>>. Acesso em: 10 ago. 2017.

WAZLAWICK, R. S. **Engenharia de Software: conceitos e práticas**. Rio de Janeiro: Elsevier, 2013.

ZAMBON, F.; BEHLAU, M. **A Voz do Professor Aspectos do Sofrimento Vocal Profissional**. São Paulo: Baseado, 2009. 28 p.

ANEXO

ITERAÇÃO 03

USO 03 –Desenvolvimento da API do sistema

| TESTES DE ACEITAÇÃO | | | | | STATUS | |
|---------------------|--|-------------|---------------------|------------|--------|--|
| TA3.1 | Definição dos dados que serão fornecidos pela API do sistema | | | | | |
| TA3.2 | Criação dos <i>endpoints</i> a serem fornecidos pela API | | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status | |
| A3.1 | Definição dos dados que serão fornecidos pela API do sistema | Tarciana | | | | |
| A3.2 | Definir os <i>endpoints</i> a serem fornecidos pela API | Tarciana | | | | |
| A3.3 | Criar os métodos de acesso à API | Tarciana | | | | |

Quadro 17: Desenvolvimento da API do Sistema

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 04

USO 04 – Implementação testes e validação das funcionalidades informações sobre anatomia do aparelho vocal.

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA4.1 | Usuário buscar informações sobre anatomia do aparelho vocal (dados escritos, vídeos e imagens) no APLICATIVO. | | | | |
| TA4.2 | Buscar palavra contida no glossário (a palavra pesquisada deve ser exibida) | | | | |
| TA4.3 | Buscar palavra não existente no glossário (a palavra pesquisada não pode ser exibida) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A4.1 | Desenvolver busca sobre anatomia do aparelho vocal | Tarciana | | | |
| A4.2 | Implementar tela inicial para o glossário | Tarciana | | | |
| A4.3 | Desenvolver o bd das palavras contidas no glossário | Tarciana | | | |
| | Desenvolver o mecanismo de busca das palavras contidas no glossário | Tarciana | | | |

Quadro 18: Implementação, Testes e Validação das Funcionalidades informações sobre anatomia do aparelho vocal.

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 05

USO 05 – Implementação testes e validação das funcionalidades informações sobre os inimigos da voz na API

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA5.1.1 | Buscar informações sobre inimigo número 1 | | | | |
| TA5.2.3 | Buscar informações sobre inimigo número n | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A5.1 | Desenvolver busca para informações sobre inimigo número 1 | Tarciana | | | |
| A5.2 | Desenvolver busca para informações sobre inimigo número N | Tarciana | | | |
| A5.3 | Desenvolver endpoint fornecedor dos dados anteriores | Tarciana | | | |

Quadro 19: Implementação, Testes e Validação das Funcionalidades: informações sobre os inimigos da voz na API.

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 06

USO 06 – Implementação testes e validação das funcionalidades cuidados com a voz na API

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|--|-------------|---------------------|------------|--------|
| TA6.1 | Buscar informações sobre orientações nos cuidados com a voz (dados escritos, vídeos e imagens) | | | | |
| TA6.2 | Visualizar conteúdo total do fórum de discussões | | | | |
| TA6.2.1 | Submeter contribuição válida para o fórum (atualizar lista de contribuições válidas do fórum) | | | | |
| TA6.2.2 | Submeter contribuição inválida para o fórum (a contribuição não deve ser adicionada ao fórum) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A6.1 | Desenvolver busca para orientações nos cuidados com a voz. | Tarciana | | | |
| A6.2 | Implementar a tela inicial do fórum | Tarciana | | | |
| A6. | Implementar o mecanismo de contribuição do fórum | Tarciana | | | |
| A6.6 | Implementar restrições de segurança no fórum. | Tarciana | | | |

Quadro 20: Implementação, Testes e Validação das Funcionalidades: cuidados com a voz na API

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 07

USO 07 – Implementação testes e validação das funcionalidades voz saudável, comunicação em dia na API

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|--|-------------|---------------------|------------|--------|
| TA7.1 | Buscar informações sobre mitos e verdades sobre os cuidados com a voz (dados escritos, vídeos e imagens) | | | | |
| TA7.2 | Buscar informações sobre higiene vocal (dados escritos, vídeos e imagens) | | | | |
| TA7.3 | Buscar informações sobre as FAQs | | | | |
| TA7.3.1 | Usuário ler e entender as FAQs | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A7.1 | Desenvolver busca para mitos e verdades sobre os cuidados com a voz | Tarciana | | | |
| A7.2 | Desenvolver o mecanismo de apresentação das FAQ'S | Tarciana | | | |
| A7.3 | Pesquisar material para as FAQs | Tarciana | | | |
| A7.3.1 | Produzir material para as FAQs | Tarciana | | | |
| A7.3.2 | Implementar tela inicial das FAQs | Tarciana | | | |

Quadro 21: Implementação, Testes e Validação das Funcionalidades: voz saudável, comunicação em dia na API

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 08

USO 08 – Implementação testes e validação das funcionalidades voz profissional na API

| TESTES DE ACEITAÇÃO | | | | | STATUS | |
|---------------------|--|---|---------------------|------------|--------|--|
| TA8.1 | | Buscar informações sobre o papel da voz nas profissões (dados escritos, vídeos e imagens) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status | |
| A8.1 | Desenvolver busca para informações sobre o papel da voz nas profissões | Tarciana | | | | |
| A8.2 | Criação dos <i>endpoints</i> que fornecem os dados do papel da voz e <i>endpoints</i> de atualização desses dados. | Tarciana | | | | |
| A8.3 | Integração da API do projeto à API de vídeos fornecida pelo <i>YouTube</i> | Tarciana | | | | |

Quadro 22: Implementação, Testes e Validação das Funcionalidades: voz profissional na API

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 09

USO 09 – Implementação testes e validação das funcionalidades a funcionalidade manter profissionais na API

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|--|-------------|---------------------|------------|--------|
| TA9.1 | Cadastrar um profissional informando todos os dados (cadastro efetuado com sucesso). | | | | |
| TA9.2 | Cadastrar um profissional sem informar todos os dados (cadastro não efetuado) | | | | |
| TA9.3 | Fazer alteração no endereço de trabalho (mudança) | | | | |
| TA9.4 | Fazer busca pelo endereço de trabalho (endereço deve ser exibido) | | | | |
| TA9.5 | Fazer alteração na especialidade do profissional (cursos extras, aperfeiçoamento, especialização...) | | | | |
| TA9.6 | Excluir cadastro do profissional (por motivo de morte ou outro) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A9.1 | Desenvolver o sistema de cadastro dos profissionais integrado ao banco de dados. | Tarciana | | | |
| A9.2 | Integrar os dados dos profissionais à API do Google MAPs dados. | Tarciana | | | |
| A9.3 | Desenvolver os métodos de exclusão bem como os <i>endpoints</i> de comunicação da API | Tarciana | | | |

Quadro 23: Implementação, Testes e Validação das funcionalidades: a funcionalidade manter profissionais na API

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 10

USO 10 – Implementação testes e validação das funcionalidades guia simples de ajuda

| TESTES DE ACEITAÇÃO | | | | | STATUS | |
|---------------------|---|--|---------------------|------------|--------|--|
| TA10 | | Procurar informações diversas no guia de ajuda | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status | |
| A10.1 | Produzir informações sobre o aplicativo | Tarciana | | | | |
| A10.2 | Exportar o informativo via API | Tarciana | | | | |

Quadro 24: Implementação, Testes e Validação das Funcionalidades: guia simples de ajuda

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 11

USO 11– Prototipação das telas desenvolvidas estaticamente no IONIC

| TESTES DE ACEITAÇÃO | | | | | STATUS | |
|---------------------|--|--|---------------------|------------|--------|--|
| TA11 | | Criação do ambiente de desenvolvimento em IONIC | | | | |
| TA11.1 | | Prototipação das telas estáticas em componentes Angular a serem gerenciados pelo IONIC | | | | |
| TA11.2 | | Fluxo de telas no sistema utilizando mocks em para fornecimento dos dados | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status | |
| A11.1 | Criação do ambiente de desenvolvimento | Tarciana | | | | |

| | | | | | |
|-------|---|----------|--|--|--|
| | em IONIC | | | | |
| A11.2 | Prototipação das telas estáticas em componentes Angular a serem gerenciados pelo IONIC | Tarciana | | | |
| A11.3 | fluxo de telas no sistema utilizando <i>mocks</i> para fornecimento dos dados. o fluxo será feito com a ferramenta Marvel | Tarciana | | | |

Quadro 25: Prototipação das Telas Desenvolvidas Estaticamente no IONIC

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 12

USO 12 – Integração entre *frontend* e *backend* da funcionalidade informações sobre anatomia do aparelho vocal

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|--|-------------|---------------------|------------|--------|
| TA12 | Buscar informações sobre anatomia do aparelho vocal (dados escritos, vídeos e imagens) | | | | |
| TA12.1 | Buscar palavra contida no glossário (a palavra pesquisada deve ser exibida) | | | | |
| TA12.2 | Buscar palavra não existente no glossário (a palavra pesquisada não pode ser exibida) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A12.1 | Desenvolver busca para informações sobre anatomia do aparelho vocal | Tarciana | | | |
| A12.2 | Desenvolver o bd das palavras que estarão contidas no | Tarciana | | | |

| | | | | | |
|--------------|---|----------|--|--|--|
| | glossário | | | | |
| A12.3 | Desenvolver o mecanismo de busca das palavras contidas no glossário | Tarciana | | | |

Quadro 26: Integração entre *Frontend* e *Backend* da Funcionalidade: informações sobre anatomia do aparelho vocal

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 13

USO 13 – Integração entre *frontende backend* da funcionalidade informações sobre os inimigos da voz

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA13 | Buscar informações sobre inimigo número 1(dados escritos, vídeos e imagens) | | | | |
| TA13.1 | Buscar informações sobre inimigo número n(dados escritos, vídeos e imagens) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A13.1 | Desenvolver mecanismo de busca para inimigo número 1 | Tarciana | | | |
| A13.2 | Desenvolver mecanismo de busca para inimigo N | Tarciana | | | |

Quadro 27: Integração entre *Frontende Backend* da Funcionalidade: informações sobre os inimigos da voz

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 14

USO 14 – Integração *frontend* e *backend* da funcionalidade informações sobre os cuidados com a voz

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA14 | Buscar informações sobre orientações nos cuidados com a voz (dados escritos, vídeos e imagens) | | | | |
| TA14.1 | Visualizar conteúdo total do fórum de discussões | | | | |
| TA14.2 | Submeter contribuição válida para o fórum (atualizar lista de contribuições válidas no fórum) | | | | |
| TA14.3 | Submeter contribuição inválida para o fórum (a contribuição não deve ser adicionada ao fórum) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A14.1 | Desenvolver busca sobre orientações nos cuidados com a voz | Tarciana | | | |
| A14.2 | Implementar interface gráfica do fórum de discussões | Tarciana | | | |
| A14.3 | Implementar funcionalidade de criação de tópico no fórum junto ao banco de dados e integrado à interface gráfica; | Tarciana | | | |
| A14.4 | implementar funcionalidade de comentários de tópico junto ao banco de dados e integrado à interface gráfica; | Tarciana | | | |

Quadro 28: Integração *Frontend* e *Backend* da Funcionalidade: informações sobre os cuidados com a voz

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 15

USO 15 – Integração entre *frontend* e *backend* da funcionalidade informações sobre voz saudável, comunicação em dia

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA15 | Buscar informações sobre mitos e verdades nos cuidados com a voz (dados escritos, vídeos e imagens) | | | | |
| TA15.1 | Buscar informações sobre higiene vocal (dados escritos, vídeos e imagens) | | | | |
| TA15.2 | Buscar informações sobre as FAQs | | | | |
| TA15.3 | Usuário ler e entender as FAQs | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A15.1 | Desenvolver busca de informações sobre mitos e verdades nos cuidados com a voz | Tarciana | | | |
| A15.2 | Desenvolver busca sobre higiene vocal | Tarciana | | | |
| A15.3 | Pesquisar material para as FAQs | Tarciana | | | |
| A15.4 | Produzir material para as FAQs | Tarciana | | | |
| A15.5 | Implementar tela inicial das FAQs | Tarciana | | | |

Quadro 29: Integração entre *Frontend* e *Backend* da Funcionalidade: informações sobre voz saudável, comunicação em dia

Fonte: Elaborado pela autora (2017).

ITERAÇÃO 16

USO 16 – Integração entre *frontend* e *backend* da funcionalidade informações sobre voz profissional

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|--|---|---------------------|------------|--------|
| TA16 | | Buscar informações sobre o papel da voz nas profissões (dados escritos, vídeos e imagens) | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A16.1 | Desenvolver busca de informações sobre o papel da voz nas profissões | Tarciana | | | |

Quadro 30: Integração entre *Frontend* e *Backend* da Funcionalidade: informações sobre voz profissional
Fonte: Elaborado pela autora (2017).

ITERAÇÃO 17

USO 17 – Integração entre *frontend* e *backend* da funcionalidade manter profissionais

| TESTES DE ACEITAÇÃO | | STATUS |
|---------------------|---|-----------------|
| TA17 | Cadastrar um profissional informando todos os dados (cadastro efetuado com sucesso) | Ver mais testes |
| TA17.1 | Cadastrar um profissional sem informar todos os dados (cadastro não efetuado) | |
| TA17.2 | Fazer alteração no endereço de trabalho (mudança) | |

| TA17.3 | Fazer busca pelo endereço de trabalho (endereço deve ser exibido) | | | | |
|-----------|--|-------------|---------------------|------------|--------|
| TA17.4 | Fazer alteração na especialidade do profissional (cursos extras, aperfeiçoamento, especialização...) | | | | |
| TA17.5 | Excluir cadastro do profissional (por motivo de morte ou outro) | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A17.1 | Desenvolver mecanismo de cadastro dos profissionais | Tarciana | | | |

Quadro 31: Integração entre *Frontend* e *Backend* da Funcionalidade: manter profissionais
Fonte: Elaborado pela autora (2017).

ITERAÇÃO 18

USO 18 – Integração entre *frontend* e *backend* da funcionalidade guia simples de ajuda.

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA18 | Procurar informações no guia simples de ajuda | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A18.1 | Desenvolver guia de ajuda ao usuário | Tarciana | | | |
| A18.2 | Fornecer dados do guia via API | | | | |

Quadro 32: Integração entre *Frontend* e *Backend* da Funcionalidade: guia simples de ajuda.
Fonte: Elaborado pela autora (2017).

ITERAÇÃO 19

USO 19 – Validação das integrações realizadas

| TESTES DE ACEITAÇÃO | | | | STATUS | |
|---------------------|---|-------------|---------------------|------------|--------|
| TA19 | Testes de aceitação | | | | |
| TA19.1 | Avaliação final da Usabilidade do sistema | | | | |
| TA19.2 | Testar Versão Beta gerada. | | | | |
| Atividade | Descrição | Responsável | Estimativa de tempo | Tempo real | Status |
| A19.1 | Realizar <i>deploy</i> para teste da aplicação | Tarciana | | | |
| A19.2 | <i>Refatorar</i> e corrigir <i>error</i> | Tarciana | | | |
| A19.3 | Gerar Versão Beta teste gerada palavra pesquisada não pode ser exibida) | Tarciana | | | |

Quadro 33: Validação das Integrações Realizadas

Fonte: Elaborado pela autora (2017).