



UEPB

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

MARIA RAFAELLY FERREIRA ALVES

**UMA PROPOSTA DE METODOLOGIA DE SOFTWARE
EDUCACIONAL PARA O ENSINO DE MATEMÁTICA EM
LIBRAS**

**PATOS – PB
2019**

MARIA RAFAELLY FERREIRA ALVES

**UMA PROPOSTA DE METODOLOGIA DE SOFTWARE
EDUCACIONAL PARA O ENSINO DE MATEMÁTICA EM
LIBRAS**

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

Área de concentração: Metodologia de software

Orientadora: Prof.^a Amanda Mayara Sobral Rodrigues

**PATOS – PB
2019**

É expressamente proibido 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 do trabalho.

A474p Alves, Maria Rafaelly Ferreira.
Uma proposta de metodologia de software educacional para o ensino de matemática em LIBRAS [manuscrito] / Maria Rafaelly Ferreira Alves. – 2019.
51 f.: il. colorido.

Digitado.
Trabalho de Conclusão de Curso (Graduação em Computação) – Universidade Estadual da Paraíba, Centro de Ciências Exatas Sociais e Aplicadas, 2019.
“Orientação: Profa. Lic. Amanda Mayara Sobral Rodrigues, Coordenação do Curso de Computação - CCEA”.

1. Software Educacional. 2. Metodologias de Softwares. 3. Educação de Surdos. 4. Ensino de Matemática. I. Título.

21. ed. CDD 005.11


Maria Rafaelly F. Alves

Uma proposta de metodologia de software educacional para o ensino de matemática em libras


Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Ciências da Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Bacharel em Ciências da Computação.

Aprovado em 18/06/2019


BANCA EXAMINADORA



Prof. . Amanda Mayara S. Rodrigues
(Orientadora)



Prof. Esp. Sérgio Morais Cavalcante Filho
(Examinador)



Prof. M^a. Lidiane Rodrigues Campêlo da Silva
(Examinadora)

Aos meus pais, pela dedicação, carinho e amizade, DEDICO.

AGRADECIMENTOS

Aos meus pais, que apesar de todas as dificuldades, me ajudaram na realização do meu sonho.

A todos os funcionários da Universidade Estadual da Paraíba por todo apoio e por proporcionarem um ambiente propício para o desenvolvimento do meu trabalho de conclusão de curso.

À professora Amanda Mayara Sobral Rodrigues pelas leituras sugeridas ao longo dessa orientação e pela dedicação.

Aos professores do Curso de Computação da UEPB, que contribuíram ao longo desse tempo, por meio das disciplinas e debates, para o desenvolvimento desta pesquisa.

A Nilma Lucia Gomes de Lucena, pela sua dedicação na inclusão dos alunos deficientes na rede municipal de ensino da cidade de Patos-PB.

Aos colegas de classe pelos momentos de amizade e apoio.

Por fim, sou grata a todos que direta ou indiretamente contribuíram para a realização desse projeto.

RESUMO

Esta pesquisa teve por objetivo propor uma metodologia de software educacional para o ensino da matemática a pessoas surdas. A escolha da temática se deu a partir da vivência em sala de aula da autora, quando foi necessário utilizar aplicativos matemáticos para a adaptação dos conteúdos no ensino fundamental I para a aluna surda. Verificando de forma empírica a falta de aplicativos para o ensino da matemática em libras. Assim, uma dos objetivos específicos foi verificar os estudos existentes sobre o ensino de matemática a pessoas surdas com o uso de softwares educacionais. A problemática levantada neste estudo é como aperfeiçoar o desenvolvimento desses softwares educacionais matemáticos com foco na educação de surdos através da Linguagem Brasileira de Sinais. O segundo passo da pesquisa, buscou-se verificar as metodologias de desenvolvimento de software utilizadas na criação de software educacionais. Foi possível notar a carência de metodologias de softwares educacionais. Com essas pesquisas foi identificada a necessidade da realização de estudos voltados a melhorias para o processo de desenvolvimento de softwares para o ensino de matemática em libras. O estudo realizado por Benitti et al. (2005) propõe uma equipe multidisciplinar envolvida no processo de desenvolvimento de softwares educacionais. Como resultado da pesquisa, foi feito um comparativo entre as metodologias Extreme programming (XP) e Scrum e easYProcess (YP) e o modelo em cascata, a fim de verificar qual metodologia cumpri o máximo das etapas propostas por Benitti et al. (2005). Como conclusão, foi apresentada a proposta da metodologia de software, entretanto, para efetividade das etapas descritas na metodologia, é necessário a realização estudos de caso, a fim de verificar a sua contribuição para o desenvolvimento de softwares educacionais matemáticos em libras.

Palavras Chaves: Software educacional. Metodologias de softwares. Educação de surdos. Ensino de Matemática.

ABSTRACT

This research aimed to propose an educational software methodology for teaching mathematics to deaf people. The choice of the theme was based on the experience in the author's classroom, when it was necessary to use mathematical applications to adapt the content in elementary school I to the deaf student. Checking empirically the lack of applications for teaching math in pounds. Thus, one of the specific objectives was to verify the existing studies on the teaching of mathematics to deaf people with the use of educational software. The problem raised in this study is how to improve the development of these mathematical educational software focused on the education of the deaf through the Brazilian Language of Signals. The second step of the research was to verify the methodologies of software development used in the creation of educational software. It was possible to notice the lack of methodologies of educational software. These studies have identified the need to carry out studies aimed at improving the software development process for the teaching of mathematics in pounds. The study by Benitti et al. (2005) proposes a multidisciplinary team involved in the process of developing educational software. As a result of the research, a comparison was made between Extreme programming (XP) and Scrum and easYProcess (YP) and cascade models, in order to verify which methodology fulfilled the maximum steps proposed by Benitti et al. (2005). In conclusion, the proposal of the software methodology was presented, however, for the effectiveness of the steps described in the methodology, it is necessary to carry out case studies in order to verify its contribution to the development of mathematical educational software in pounds.

Keywords: Educational software. Software methodologies. Education of the deaf. Mathematics Teaching.

LISTA DE FIGURAS

Figura 1 – Sinal referente à multiplicação.....	22
Figura 2 – Sinal de duas vezes que indica a multiplicação em dobro.	22
Figura 3 – Representação do modelo em cascata	25
Figura 4 – Representação dos papéis na metodologia YP.....	28
Figura 5 – Visão Geral da metodologia proposta por Benitti et al. (2005)	29
Figura 6 – Características da qualidade de software segundo a NBR ISO/IEC 9126-1.....	31
Figura 7 – Ciclo de elaboração	43

SUMÁRIO

1.	INTRODUÇÃO	12
1.1	PROBLEMÁTICA.....	13
1.2	JUSTIFICATIVA.....	14
1.3	OBJETIVOS.....	16
1.3.1	OBJETIVO GERAL.....	16
1.3.2	OBJETIVOS ESPECÍFICOS	16
1.4	METODOLOGIA.....	16
1.5	ESTRUTURA DO TRABALHO.....	18
2.	FUNDAMENTAÇÃO TEÓRICA	19
2.1	HISTÓRIA DOS SURDOS.....	19
2.1.1	SURDOS NO BRASIL	19
2.2	LINGUAGEM BRASILEIRA DE SINAIS (LIBRAS).....	20
2.3	MATEMÁTICA E LIBRAS.....	21
2.4	IMPORTÂNCIA DA COMPUTAÇÃO NA EDUCAÇÃO DE SURDOS.....	23
2.5	METODOLOGIA DE SOFTWARE.....	24
2.5.1	METODOLOGIAS TRADICIONAIS	25
2.5.2	METODOLOGIAS ÁGEIS.....	26
2.5.3	METODOLOGIAS PARA O DESENVOLVIMENTO DE SOFTWARES EDUCACIONAIS	29
2.6	CARACTERÍSTICAS DE UM SOFTWARE PARA O ENSINO DA MATEMÁTICA.....	31
2.7	COMO ESCOLHER UMA METODOLOGIA?.....	33
2.8	ETAPAS PARA ELABORAR E VALIDAR UMA METODOLOGIA DE SOFTWARE	34
3.	RESULTADOS E DISCUSSÕES	35
3.1	INVESTIGANDO CONTRIBUIÇÕES PARA O DESENVOLVIMENTO DA METODOLOGIA DE SOFTWARE	35
3.2	DESCRIÇÃO GERAL DA METODOLOGIA DE SOFTWARE MATEMÁTICO COM FOCO EM LIBRAS.....	36
3.2	ETAPAS DA METODOLOGIA.....	37
4.	CONCLUSÕES E CONSIDERAÇÕES FINAIS	47

4.1	LIMITAÇÕES.....	48
4.2	TRABALHOS FUTUROS.....	48
	REFERÊNCIAS	49

1. INTRODUÇÃO

O avanço da informática proporcionou um horizonte de novas tecnologias que auxiliam o ensino e aprendizado em diversas áreas do conhecimento. Através de softwares educacionais o professor e o aluno são inseridos em um ambiente interativo e dinâmico. Entretanto, para que esses softwares sejam satisfatórios é necessário que eles apresentem uma boa qualidade de interface.

Segundo Souza e Silva (2009, p. 02) “a maioria dos usuários entende a interface como o próprio software, devido a não terem o mesmo conhecimento que os profissionais desenvolvedores”. A interface deve ser desenvolvida pensando no usuário. Deve apresentar usabilidade, confiabilidade e segurança no tratamento de informações.

O projeto de interface depende da metodologia aplicada no desenvolvimento do software. Metodologia de desenvolvimento de software é um conjunto de ações que a equipe desenvolvedora deve seguir para a produção de um software. Os softwares desenvolvidos para a educação, em muitos casos, não tem uma metodologia adequada para seu desenvolvimento e muitos acabam não sendo aceito pelo o usuário final, o aluno.

Outro aspecto que requer atenção dos profissionais da computação é a falta de ferramentas pedagógicas para auxiliar a educação de surdos em sala de aula. Os surdos são inseridos em um ambiente escolar preparado para educar ouvintes o que causa obstáculos de comunicação e compreensão dos conteúdos abordados.

Além das barreiras de comunicação enfrentadas pelos surdos um dos principais obstáculos é o ensino da matemática. A linguagem matemática devido a sua complexidade , pode causa barreiras de aprendizado, fazendo com que conteúdos simples não sejam entendidos pelos alunos surdos.

Os conteúdos matemáticos não são entendidos pelos alunos surdos, devido a uma série de questões. Apresentam dificuldades de compreender o problema enunciado nas questões e a postura tradicional de ensino.

A vivência pessoa da autora, como profissional Cuidador, no município de Patos- PB em 2018 foi o ponto de partida para a presente pesquisa. Profissional Cuidador tem a função de auxiliar alunos deficientes em sala de aula adaptando os conteúdos junto aos professores, para garantir a inclusão e o aprendizado do aluno. Ao tentar adaptar o conteúdo matemático para a aluna surda, que a autora acompanhava, notou-se a falta de aplicativos voltados para o ensino de matemática a surdos.

Atualmente, existem vários aplicativos de alfabetização com foco na língua portuguesa,

mas para o ensino da matemática é escasso. Para Bassani et al. (2006, p.01) “um dos aspectos que chama a atenção no desenvolvimento de software educativo é a carência de uma metodologia específica para projetá-los e desenvolvê-los”. Assim, a presente pesquisa discutirá a importância de desenvolver uma metodologia de software educacional matemático para a educação de surdos .

1.1 PROBLEMÁTICA

A carência de metodologia de desenvolvimento de softwares educativos para surdos é um tema pouco discutido. É importante desenvolver uma ferramenta computacional que auxilie o aluno e o professor em sala de aula.

A inclusão em sala de aula de alunos surdos acontece de forma lenta devido à falta de qualificação dos profissionais e a falta de acesso a recursos didáticos adequados. As ferramentas computacionais ampliam o acesso a informações em diferentes níveis de aprendizado.

Aplicativos educacionais são uma grande aposta para melhorar o ensino. O Levantamento de requisitos deve ser feito de forma minuciosa. Para o desenvolvimento desses aplicativos, segundo Marçal et al. (2010, p. 2) “devem ser enfatizados os aspectos pedagógicos e metodológicos”.

Levantamento de requisitos é a parte mais importante do processo de desenvolvimento de um software, consiste em verificar o desejo e as regras de negócio do cliente. Existem diversas técnicas de levantamento de requisitos, fica a critério da empresa desenvolvedora decida a melhor técnica para o projeto em desenvolvimento. Dentre essas técnicas podemos destacar três delas: *Brainstorming*, entrevista e questionário.

Na técnica *Brainstorming* são realizadas reuniões que permitem que todas as pessoas envolvidas no projeto informem inúmeras ideias. As pessoas envolvidas nesse processo são selecionadas pensando em suas contribuições para projeto. O líder da sessão deve direcionar a reunião para garantir que os participantes do projeto entenda o conceito de *Brainstorming* e tragam boas ideias.

A entrevista é uma técnica tradicional e mais simples de ser usada. O entrevistador utiliza perguntas objetivas e claras para que o entrevistado possa expor suas ideias sobre o projeto. Já a utilização de questionários é indicada quando diferentes usuários estão separados geograficamente.

Softwares voltados ao ensino de matemática permitem que conceitos sejam explorados de forma atrativa pelo aluno. Na educação de surdos a utilização desses softwares é uma

alternativa de ensino que visa garantir o aprendizado dos alunos surdos. Entretanto, são poucos os softwares voltados para o ensino da matemática a surdos.

Outro fator importante para o sucesso do projeto é a escolha de uma metodologia de desenvolvimento de software adequada para o projeto. Segundo (CHAVES, 2017,p. 32) “a produção de um software educativo requer a utilização de metodologias que orientam e definam os processos para concepção desse tipo de produto especializado”.

Dada à importância de desenvolvimento de metodologias para a implementação de softwares educacionais voltados a educação de surdos em Libras (Linguagem Brasileira de sinais), esse trabalho busca investigar as necessidades educacionais de pessoas surdas. A pesquisa terá a seguinte problemática: *Como desenvolver uma metodologia para o desenvolvimento de um software que auxilie o ensino da matemática a pessoas surdas em um mundo de ouvintes?*

1.2 JUSTIFICATIVA

A comunicação nos torna sociáveis e capazes de obter e trocar informações com a sociedade. Segundo Carneiro (2013, p.29) “a comunicação sempre foi e será um dos pilares fundamentais de uma sociedade, pela sua importância e impacto na construção da mesma”.

Só é possível estabelecer uma comunicação adequada se tiver um emissor, o receptor e o canal que é a linguagem utilizada para a troca de informação. Ela pode ocorrer de diversas formas: oral ou escrita, com o uso de pintura, formas geométricas ou sinais.

Existe a Língua Brasileira de sinais (Libras), linguagem utilizada para quem tem deficiência auditiva ou não se comunicarem sem a necessidade da fala. A Lei nº 10.436, de 24 de Abril de 2002, reconhece a Língua Brasileira de Sinais (Libras), mas essa lei deixa claro que essa modalidade de comunicação não poderá substituir a Língua Portuguesa.

Segundo a LEI Nº 10.436(2002, p.1) “deve ser garantido, por parte do poder público em geral e empresas de serviços públicos, apoiar o uso e difusão da Libras como meio de comunicação objetiva e de utilização corrente das comunidades surdas do Brasil”. Em complemento a esta lei, temos a lei de Nº 13.146, de julho de 2015.

A lei de Nº 13.146, de julho de 2015, institui a lei Brasileira de inclusão da pessoa com deficiência. Ela é destinada a promover e assegurar os direitos de igualdade a pessoa com deficiência cujo objetivo de incluí-las em sociedade.

O capítulo IV, do direito a educação, no artigo 28º da lei de Nº 13.146 (2015) inciso VI, afirma que devem ser realizadas “pesquisas voltadas para o desenvolvimento de novos métodos

e técnicas pedagógicas, de materiais didáticos, de equipamentos e de recursos de tecnologia assistiva(LEI, N° 13.146 , 2015,P.4) ”.

Bertoli (2012, p.2) diz que “a principal dificuldade, além da comunicação com surdos, é adaptar a linguagem matemática”. Essa dificuldade, sentida pelos docentes, acaba acarretando problemas significativos para o aluno surdo que apresenta dificuldade de efetuar simples operações matemáticas.

Segundo Lobato e Noronha (2013, p.6), “é necessário viabilizar estímulos linguísticos e metodológicos no espaço escolar que possibilitem ao aluno surdo minimizar os atrasos e dificuldades em Matemática detectada em todos os níveis de ensino (...)”.

Pensar em formas de utilizar recursos tecnológicos para melhorar o ensino da matemática para surdos que proponha a inclusão dentro e fora da sala de aula é de suma importância. Para isso, é necessário criar ferramentas pedagógicas para o ensino de matemática em libras sobre o olhar do surdo.

O engenheiro paraense Walter Júnior desenvolveu um jogo educacional disponível para dispositivos móveis cujo nome é Matematicando. O aplicativo Matematicando realiza operações básicas de soma, subtração, divisão e multiplicação. Segundo Dourado et al. (2017, p.5),

O objetivo deste jogo é mostrar para os alunos que a Matemática pode ser trabalhada de várias formas e que o aluno surdo tem a capacidade de aprender juntamente com todos, em um só ambiente, sabemos que o professor encontra muitas dificuldades com relação ao ensino das quatro operações, este aplicativo vem como um auxílio para o desempenho destes alunos neste conteúdo.

Assim, como o aplicativo matematicando, os softwares voltados à educação de surdos devem favorecer o aprendizado do aluno. O software educacional além de atingir seu objetivo principal, que é o benefício e incentivo do aluno, proporcionará um apoio pedagógico para o professor, de modo que, esse apoio traga benefícios de inclusão junto à sociedade e incluindo-o também ao mundo tecnológico.

Para tanto, a pesquisa também surgiu de uma indagação da autora na sua vivência dentro de sala de aula. Ao necessitar de recursos que pudessem auxiliar o aprendizado do aluno surdo sobre os conteúdos abordados sobre matemática, verificou a carências de ferramentas pedagógicas voltadas para alunos surdos nos anos iniciais do Ensino Fundamental. Pois em muitos casos os alunos surdos nesse período ainda estão em processo de aprendizado em Libras.

Segundo Alves (2016, p. 2), “a matemática nos anos iniciais do fundamental é de suma importância para os alunos, pois ela desenvolve o pensamento lógico e é essencial para construção de conhecimentos em outras áreas”. Se ela for educada de forma correta nas fases

iniciais a criança não terá problemas entender outros conceitos matemáticos.

Conforme o que foi exposto, deve-se pensar em meio de aperfeiçoar o desenvolvimento dos aplicativos matemáticos em libras para os anos iniciais do fundamental. Assim, é importante a busca e elaboração de metodologias de desenvolvimento de software educacionais voltados ao ensino de matemáticas a pessoas surdas.

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

Propor uma metodologia de desenvolvimento de software educacional voltada ao ensino de matemática a pessoas surdas.

1.3.2 OBJETIVOS ESPECÍFICOS

- Verificar os estudos existentes sobre o ensino de libras para pessoas surdas usando softwares educacionais;
- Realizar um levantamento bibliográfico sobre as metodologias utilizadas no desenvolvimento dos softwares educacionais;
- Verificar os estudos existentes sobre softwares matemáticos para o ensino a pessoas surdas;
- Investigar as maiores dificuldades do ensino da matemática para alunos surdos;
- Propor uma metodologia para criação de softwares educacionais voltados a pessoas surdos para o ensino de matemática no fundamental I, do 1º ao 5º ano;

1.4 METODOLOGIA

O presente estudo é de cunho bibliográfico, cujo objetivo é procurar explicar e discutir a temática abordada através de revistas, livros e artigos. A principal vantagem da pesquisa bibliográfica reside no fato de permitir ao investigador a cobertura de uma gama de fenômenos muito mais ampla do que aquela que poderia pesquisar diretamente (GIL, 2002, p. 50).

As questões de pesquisa foram:

- Como deve ser o estudo da matemática para o ensino fundamental I do 1º ao 5º por pessoas com deficiência auditiva?;
- Como deve ser um software educacional para surdos?;

- Quais as metodologias de desenvolvimento de softwares existentes?
- Existe metodologia de desenvolvimento de software educacional com foco na linguagem brasileira de sinais?.

Foi utilizadas bases de dados específicos com grau de relevância para a área de estudo. As pesquisas devem priorizar o idioma português, pois libras é a linguagem Brasileira de sinais e cada país possui uma linguagem de sinais. As principais fontes de pesquisas são:

- Google scholar;
 - Instituto Nacional de Educação de Surdos (INES);
- As *strings* de pesquisa foram:
- “Educação de surdos”, “alunos e professores”, “matemática para o ensino fundamental”, “educação e inclusão”.
 - “Linguagem brasileira de sinais(libras)”, “praticas pedagógicas para o ensino de matemática para alunos com deficiência auditiva”, “tecnologias utilizadas em sala de aula para o ensino da matemática para alunos do 1° ao 5 ”, “aplicativos para o ensino de matemática em libra”.
 - “Metodologias para desenvolvimento de softwares educacionais”, “metodologias de desenvolvimento de softwares”, “metodologias de desenvolvimento de softwares educacionais de matemática”.

A segunda etapa da investigação é buscar a possível contribuição da computação para o aprendizado de alunos surdos, encontrando softwares direcionados ao ensino. Na terceira etapa, foi feito uma pesquisa sobre as metodologias de desenvolvimento de software, principalmente visando encontrar metodologias para o desenvolvimento de softwares educacionais.

Encontradas as dificuldades, a penúltima etapa da pesquisa, foi realizado um comparativo entre as metodologias de softwares aqui estudadas, a fim de encontrar a que melhor cumprisse com os requisitos básicos para uma boa metodologia de software voltada à educação.

A última etapa utilizou os conhecimentos adquiridos no levantamento bibliográfico e nas metodologias de softwares estudadas, para se criar uma proposta de metodologia voltada ao desenvolvimento de softwares educacionais matemáticos para a educação de surdos.

Em suma, foi realizada uma análise dos recursos disponíveis para o desenvolvimento de um software matemático para surdos, utilizando a Linguagem Brasileira de Sinais (Libras).

1.5 ESTRUTURA DO TRABALHO

Este capítulo 1, intitulado introdução, apresentou o cenário que será abordado na presente pesquisa, levantando a problemática e a justificativa do tema. Também, evidencia os objetivos da pesquisa e a metodologia que será aplicada no decorrer do estudo.

O capítulo 2 apresenta a fundamentação teórica da pesquisa, será abordada a história dos surdos no contexto geral e no Brasil, a Linguagem brasileira de sinais e sua relação com a matemática. Também, São apresentados os principais grupos de metodologias de softwares a fim de verificar a sua importância.

O capítulo 3 traz os resultados da pesquisa, cujo objetivo é propor uma metodologia de software educacional matemático em libras. No capítulo 4, serão mostradas as conclusões obtidas com o trabalho, suas contribuições e sugestões para trabalhos futuros. E por fim, as referências utilizadas na pesquisa.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os principais autores de estudos que contribuíram para a compreensão das necessidades educacionais dos surdos no Brasil. São apresentados os fatos históricos envolvendo a pessoa surda, a forma como ele se comunica e como deve ser desenvolvida uma metodologia de software para a educação de surdo que vise propor a inclusão em sala de aula.

2.1 HISTÓRIA DOS SURDOS

Não havia a distinção entre pessoas surdas e deficientes mentais, por esse motivo eles não eram incluídos na sociedade e na educação. Strobel (2006, p.247) afirmou que “na antiguidade, os sujeitos surdos eram estereotipados como ‘anormais’, com algum tipo de atraso de inteligência, devido à ausência de trabalho e pesquisas científicas desenvolvidos na área educacional”.

Apenas no século XVI que se iniciaram os estudos e debates sobre a inclusão de pessoas surdas na sociedade. Na Europa, vários estudiosos iniciavam suas pesquisas de forma independente, e foi nesse cenário que se iniciou também o debate sobre a oralidade ou a linguagem de sinal.

Em 1880 em um congresso realizado em Milão, Segundo Strobel (2006, p.247) “o uso de língua de sinais foi definitivamente banido a favor do oralismo”. Neste período muitos professores surdos foram demitidos para dar espaço a professores que utilizavam a linguagem oral.

Nesse período, pós-congresso, os professores surdos foram substituídos por professores ouvintes que chegavam a fazer papel de “fonoaudiólogos”. As práticas pedagógicas utilizadas nesse período só intensificaram os preconceitos sobre as pessoas surdas. Apenas no século XX, com avanço da medicina, que se teve a preocupação de incluir pessoas surdas no convívio social.

2.1.1 SURDOS NO BRASIL

A história de surdos no Brasil se confundia com a história da Língua Brasileira de Sinais (Libras). A história inicia-se no século XVI, período em que marcou o retrocesso da educação para pessoas surdas no mundo.

O professor surdo, o francês E. Huet, chegou em meados de 1852 ao Brasil. Ele deu

início às pesquisas sobre educação de surdos e enviou um relatório ao imperador, D. Pedro II, apresentando duas propostas para o governo para ajudar na criação do primeiro colégio para a educação de surdos e mudos do Brasil.

A escola para surdos passou a funcionar em 1º de Janeiro de 1856, nas dependências do colégio de M. de Vassimon, no modelo privado (ROCHA, 2008, p.30).

Huet enviou uma petição, em abril de 1855, para a câmara dos deputados com o objetivo de criar do Instituto Imperial dos surdos-mudos e obter uma contribuição de 15 contos de reis anuais. A petição só foi atendida um ano e meio depois com a lei 939 de 26 de setembro de 1857, data da fundação do INES, Instituto Nacional de Educação de Surdos.

Apenas após quase um século e meio da criação do INES é que libras foi reconhecida como a Língua Brasileira de Sinais com a Lei nº 10.436, de 24 de Abril de 2002, que foi regulamentada pelo decreto de Nº 5.626/05 em Dezembro de 2005.

Juridicamente, com a lei nº 10.436 e o decreto 5.626/05, é possível compreender que para ser efetivada a educação e a inclusão de pessoas surdas é necessário pensar em tecnologias capazes de fazer um intermédio entre a linguagem de sinais à língua oral.

Apesar das investidas desde o governo de D. Pedro II até os dias atuais, a educação para surdos ainda é um grande desafio. A parte conservadora da educação ainda acredita na oralidade como parte fundamental para o aprendizado, ou por não ter formação adequada para a adaptação curricular, a falta de políticas e investimentos na educação de surdos ou simplesmente pelo comodismo de décadas.

2.2 LINGUAGEM BRASILEIRA DE SINAIS (LIBRAS)

A Linguagem Brasileira de Sinais (Libras) é a língua utilizada por pessoas com deficiências auditivas ou ouvintes para estabelecer comunicação. Ela tem influência da língua de sinais francesa, pois, como no tópico 2.1.1, um dos precursores de ensino a pessoas surdas no Brasil é um professor Francês E. Huest.

A primeira iconografia dos signos utilizados por pessoas surdas e mudas no Brasil é datada no ano de 1875, cuja sua idealização partiu do aluno do INES, Flausino José da Gama.

A estrutura gramatical de linguagem de sinais é semelhante com a língua oral. Libras possui sua gramática própria que explana os estudos: Fonológico – a arte das conversas feitas com as mãos; Morfológica – estudo dos sinais; Sintático – organização das frases; Semântico – relacionado ao significado dos sinais e o Pragmático – que é o significado ou o sentido do sinal empregado em uma frase.

Os sinais em libras matemáticos ainda são poucos, principalmente na área de geometria.

“São pouquíssimos os professores habilitados para trabalhar com os sujeitos surdos. Nas faculdades de Pedagogia, por exemplo, há especializações para uma ou outra área de deficiência (STROBEL, 2006, p.7)”.

2.3 MATEMÁTICA E LIBRAS

O aluno surdo a ser inserido em uma sala de aula direcionada ao ensino de pessoas ouvintes demora a compreender os conteúdos dados em sala de aula, isso ocorre tanto por falta ou pela precariedade da formação dos profissionais, quanto pela escassez de recursos didático-pedagógicos voltados para esse público. O processo de aprendizado do surdo, muitas vezes, se torna mais lento, pois estes não têm o estímulo adequado para aprender.

Se os alunos ouvintes já sentem dificuldade em compreender cálculos matemáticos simples, no caso do aluno surdo as dificuldades se acentuam. Nos anos iniciais Ensino Fundamental, do 1º ao 5º ano, o aluno deve ser capaz de conhecer os números, ter noção de espaço, conhecer medidas e grandezas, conhecer as operações matemáticas e ter noções de geometria, no entanto, muitas vezes esses resultados não são alcançados pelo aluno surdo.

No aprendizado significativo no ensino de matemática para um aluno com deficiência auditiva é necessário que o professor tenha domínio sobre Libras. Na prática, as escolas de ensino regular são raras os profissionais que dominam a linguagem de sinal.

É fundamental que os educadores ao fazerem seu plano de aula, organizem situações didáticas que favoreçam a aprendizagem de seus alunos surdos e levem em consideração que processo de apropriação de conhecimento deste educando passa pela exploração da competência que lhe é mais desenvolvida, que é a visual-espacial (BERTOLI, 2012, p.4-5).

As situações didáticas elaboradas no plano de aula devem ser capazes de suprir as necessidades do aluno surdo. É necessário utilizar recursos visuais para que o ensino da matemática aconteça de forma correta. Já o domínio de Libras é importante no processo de ensino.

Os autores Silva e Vasconcelos (2015, p.10) “debatem um assunto que deve ser levado em consideração, a Polissemia, ou seja, como um sinal pode assumir significados distintos”. Os exemplos dados por eles estão nas figuras 2 e 3, onde o mesmo sinal matemático possui dois significados diferentes.

Figura 1 – Sinal referente à multiplicação.



Fonte: Silva e Vasconcelos (2015, p. 9).

Figura 2 – Sinal de duas vezes que indica a multiplicação em dobro.



Fonte: Silva e Vasconcelos (2015, p. 11).

Ainda não existe uma consolidação ou uma regulamentação dos sinais em Libras para matemática (ARAUJO, 2015, p.78). Assim, além da preocupação em utilizar os sinais existentes de maneira correta, o professor e o aluno devem buscar outros meios de compreender os assuntos matemáticos.

2.4 IMPORTÂNCIA DA COMPUTAÇÃO NA EDUCAÇÃO DE SURDOS

O uso da computação no processo de aprendizagem pode se tornar uma ferramenta muito importante, propiciando uma melhor interação entre alunos com conhecimento a ser adquirido (SILVA, GUIMARÃES, 2014, p.2).

A computação aplicada à educação configura-se como um recurso didático multidisciplinar, pois ao utilizar os recursos, o aluno pode acessar diferentes plataformas de conhecimentos textuais e audiovisuais. Os softwares educacionais são uma proposta que visa incluir e dar suporte para a geração de conhecimento. Eles se mostram importantes para incluir alunos com deficiência.

Os aplicativos inicialmente desenvolvidos para os surdos, segundo Stumpf (2010, p.19) “eram construídos com os objetivos de aumentar o vocabulário, treinar a percepção da fala e estruturar a escrita dos surdos”. Os educadores ainda tinham em mente que para educar era preciso a oralidade.

Conforme os anos se passam e com a incorporação da linguagem brasileira de sinais é que foi levantada a discussão sobre o desenvolvimento de aplicativos para a educação de surdos. “Surgiram os softwares que trazem pessoas sinalizando e, no Brasil, os de escrita de sinais de Libras pelo sistema *SignWriting* (STUMPF, 2010, p.19)”.

Atualmente são encontrados diversos aplicativos voltados a alfabetização, tradução e interpretação de sinais. Um deles é o VLibras. Aplicativo móvel disponível para as plataformas IOS e android. Trata-se de um software que possui ferramentas para traduzir textos, áudios e vídeos da língua portuguesa para linguagem de sinais.

Freitas, Medeiros e Sousa (2017, P.4) realizaram um estudo com alunos surdos utilizando esse aplicativo, eles afirmaram que “durante a condução da avaliação, os alunos se mostraram empolgados com a utilização de uma nova ferramenta que pudesse fazer a inclusão de deficientes auditivos”.

É importante ressaltar que o estudo realizado por Freitas, Medeiros e Sousa (2017) só valida a importância de incluir ferramentas computacionais que proponham incluir o aluno surdo em sala de aula.

Foi encontrada, também, outra avaliação de teste utilizando o Vlibras por Reis et al. (2017). Nessa pesquisa os autores encontraram alguns problemas, como a necessidade de se ter um intérprete de libras presente para auxiliar a comunicação da equipe com os participantes do teste. Eles notaram a necessidade de realizar testes de usabilidade com pessoas surdas em futuras implementações de melhorias da ferramenta.

São inúmeras as possibilidades de utilizar recursos da informática para a educação de surdos. Entretanto, o campo de desenvolvimento desses recursos ainda é pequeno. Aplicativos voltados para o ensino da matemática são raros, bem como os aplicativos para outros componentes escolares.

2.5 METODOLOGIA DE SOFTWARE

Sistemas de software desempenham um papel importante nos dias atuais. Eles automatizam trabalhos simples ou complexos com a finalidade de criar um elo entre o usuário e a máquina.

Para desenvolver um software de qualidade a equipe desenvolvedora não deve apenas voltar seu olhar para o desenvolvimento do código, também é fundamental pensar na metodologia de desenvolvimento para que o produto final atenda as necessidades do cliente.

É necessário que a parte desenvolvedora inicie uma série de tarefas, que partem do conhecimento das necessidades do cliente/usuário até a realização de testes. A ISO 9241-210/2010 descreve como deve ser um sistema interativo focado no usuário.

A ISO 9241-210/2010 é complementar às metodologias, de forma que seus parâmetros podem beneficiar todas as partes envolvidas em busca do melhor design de interfaces, o centrado no usuário (CHAMMAS et al. ,2013, p. 3).

A engenharia de software trata todos os aspectos do processo de desenvolvimento de um software. Sommerville (2011, p.18), afirmou que “um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software”.

O processo de software (ou metodologia de software) depende de quatro atividades fundamentais. Sommerville (2011, p.18) as identifica como:

1. Especificação de software. A funcionalidade do software e as restrições ao seu funcionamento devem ser definidas.
2. Projeto e implementação de software. O software deve ser produzido para atender às especificações.
3. Validação de software. O software deve ser validado para garantir que atenda às demandas do cliente.
4. Evolução de software. O software deve evoluir para atender às necessidades de mudança dos clientes.

Existem várias metodologias de software que podem ser divididas em dois grupos. O primeiro grupo é denominado de Metodologias tradicionais e o segundo são as metodologias ágeis.

2.5.1 METODOLOGIAS TRADICIONAIS

As metodologias tradicionais foram muito utilizadas no passado. Algumas entraram em desuso devido às vantagens das metodologias ágeis, entretanto não pôde-se afirmar que elas não são mais utilizadas, tudo depende do objetivo da equipe desenvolvedora e das necessidades do software.

A principal metodologia tradicional e utiliza até hoje é o modelo em cascata. “O modelo em cascata é um exemplo de um processo dirigido a planos — em princípio, você deve planejar e programar todas as atividades do processo antes de começar a trabalhar nelas (SOMMERVILLE, 2011, p.20)”.

O objetivo principal deste do processo em cascata é desenvolver a aplicação de forma sequencial, de modo que, ao final da primeira etapa, o projeto é direcionado a segunda e assim por diante, conforme ilustrado na figura 3.

Figura 3 – Representação do modelo em cascata



Fonte: Autora, adaptada de Sommerville, 2011.

Na etapa de definição de requisitos, são descritos as restrições e serviços do sistema informados pelo cliente. Nessa fase de projeto, “o processo de projeto de sistemas aloca os requisitos tanto para sistemas de hardware como para sistemas de software, por meio da definição de uma arquitetura geral do sistema. O projeto de software envolve identificação e descrição das abstrações fundamentais do sistema de software e seus relacionamentos” (SOMMERVILLE, 2011, p.20).

Durante o estágio de implementação as unidades do software devem ser codificadas e no estágio de testes, segundo Sommerville (2011, p.21), “as unidades individuais do programa ou programas são integradas e testadas como um sistema completo para assegurar que os requisitos do software tenham sido atendidos”. Após o teste, o sistema de software é entregue ao cliente.

O último estágio, camada de manutenção, normalmente é a fase mais longa do ciclo. “A manutenção envolve a correção de erros que não foram descobertos em estágios iniciais do ciclo de vida, com melhora da implementação das unidades do sistema e ampliação de seus serviços em resposta às descobertas de novos requisitos (SOMMERVILLE, 2011, p.20)”. Cada etapa tem associada ao seu término uma documentação padrão que deve ser aprovada para que se inicie a etapa imediatamente posterior (SOARES, 2004, p.02).

Outros exemplos de metodologias tradicionais são: prototipação e modelo espiral. Ambas trazem melhorias ao modelo em cascata. Prototipação é uma versão inicial do software, Segundo Sommerville (2011, p.44) “protótipos do sistema permitem aos usuários ver quão bem o sistema dá suporte a seu trabalho”.

No modelo espiral é reconhecido de forma explícita os riscos do sistema. “Um ciclo da espiral começa com a definição de objetivos, como desempenho e funcionalidade. Em seguida, são enumeradas formas alternativas de atingir tais objetivos e de lidar com as restrições de cada um deles (Sommerville ,2011, p.47)” .

2.5.2 METODOLOGIAS ÁGEIS

O modelo de desenvolvimento ágil traz flexibilidade ao projeto. Esse modelo é baseado em ciclos iterativos e incrementais, focado em gerar melhorias contínuas para a equipe e para o projeto.

As metodologias ágeis aperfeiçoam o desenvolvimento do projeto, pois existe a busca por melhorias contínuas. Iremos discutir de forma resumida três exemplos de metodologias ágeis: a *Extreme programming* (XP) e Scrum e easYProcess (YP).

Em Extreme Programming, os requisitos são expressos como cenários (chamados de histórias do usuário), que são implementados diretamente como uma série de tarefas (SOMMERVILLE, 2011, p.44).

As principais características da metodologia XP são: *Feedback* rápido; Simplicidade nas ações; Mudanças incrementais e Aceita mudanças. Apesar das características apresentadas, segundo Souza (2007, p. 3) a metodologia XP “não deve ser aplicado a qualquer tipo de projeto. O grupo de desenvolvedores deve formar uma equipe de 2 a 10 integrantes, que devem estar por dentro de todas as fases do desenvolvimento”.

É importante destacar a presença do cliente ao longo do desenvolvimento, o que aperfeiçoa a comunicação com os desenvolvedores. O *feedback* gerado entre essa interação permite que pequenas mudanças aconteçam ao longo do desenvolvimento de forma rápida.

Esta metodologia pretende atender a cinco valores que a caracteriza. O primeiro deles é a comunicação, que é vista como fundamental para o sucesso do projeto. Como vimos anteriormente, é de suma importância a presença do cliente em todo o desenvolvimento.

Outro valor importante é o *Feedback*. Essa comunicação acontece entre cliente e desenvolvedores e entre os membros da equipe. Segundo Machado (2009, p.11) “Quanto mais *feedback* ocorrer, a chance de um projeto ter sucesso é bem maior, pois o cliente estará sempre acompanhando de perto o desenrolar do projeto e poderá dar suas sugestões em relação a ele”.

Os outros valores dessa metodologia são: Coragem; Respeito e Simplicidade. O princípio da simplicidade prioriza o desenvolvimento do artefato que foi decidido naquele momento não de outros. O princípio do Respeito deve prevalecer em todo o desenvolvimento do projeto, saber respeitar as opiniões favorece na entrega de um bom produto. Por fim, o princípio da Coragem está ligado a buscar mecanismos de segurança e ter confianças para criar soluções para erros, pois todo projeto é sujeito a falhas.

O Scrum, por outro lado, baseia-se no controle e processos com o objetivo de prevenir e controlar os riscos de um projeto. Segundo Silva, Souza e Camargo (2013, p.06) “O SCRUM trabalha com desenvolvimento incremental, dividindo seus processos em *sprints*. *Sprints* é o nome como é chamado às interações que ocorrem no SCRUM, ou seja, o período de trabalho para cada fase incremental”.

A equipe do Scrum é dividido em três papéis: *ProductOwner*, que é uma pessoa que fica responsável pelo time e pela elaboração do produto, bem como pela gerência das tarefas a serem realizadas (*ProductBacklog*); *Scrum Master*, que é o responsável por guiar a equipe desenvolvedora à utilização adequada do Scrum, bem como a criação de produtos com alto valor; equipe de desenvolvimento, que é composto por profissionais que têm como tarefa a

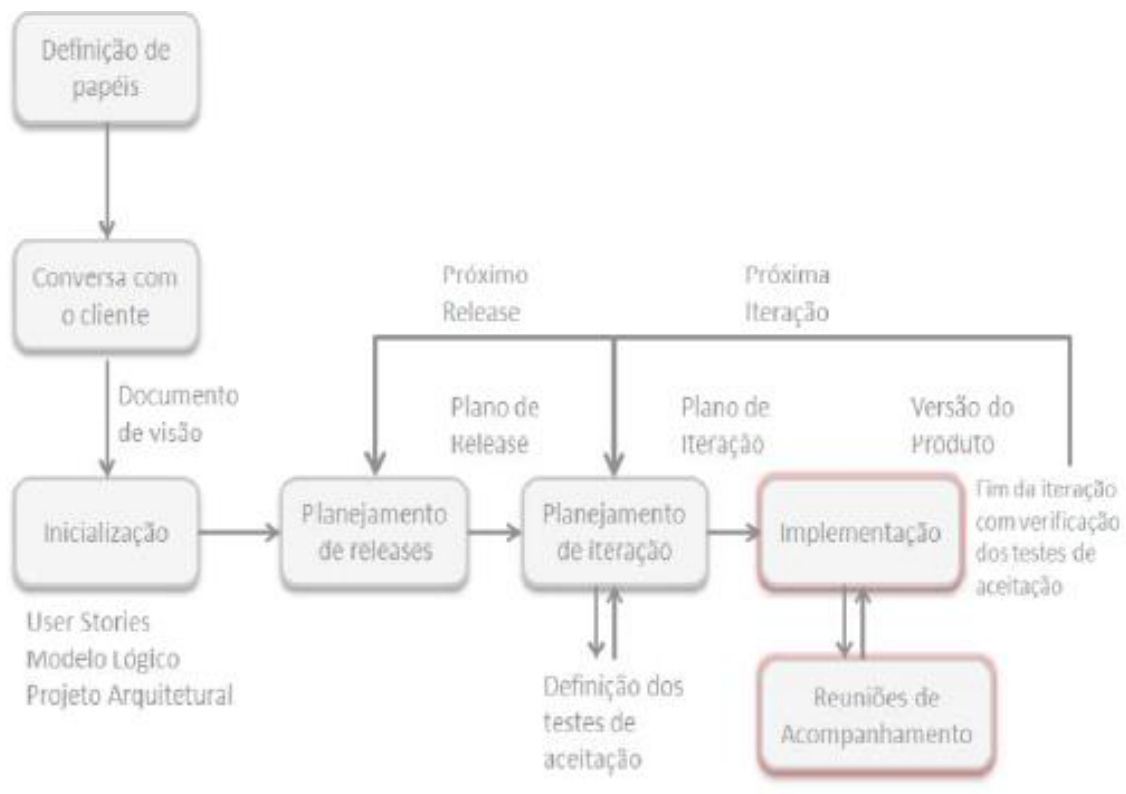
entrega de um incremento ao produto ao término de cada *Sprint* (LORIGGIO, FARIAS, MUSTARO, 2013, p.02).

Em um ambiente onde Scrum tenha sido bem implantado, os membros da equipe são estimulados e encorajados a solucionar problemas, imaginar e sugerir novas ideias e novas formas de atuar na execução de tarefas, tendo como consequência a satisfação de todos os envolvidos no processo (FERNANDES ET AL. ,2012, p.07).

A metodologia YP é um processo de software simples, desenvolvido para fins acadêmicos pela universidade Federal de Campina Grande. E apesar de ser simples, é uma metodologia completa.

De modo geral, está metodologia, possui 5 etapas. São elas: definição de papéis, conversa com o cliente, inicialização, planejamento e implementação. Em todas as etapas, a YP, sugere os papéis de cada membro da equipe. São definidos 5 papéis: Cliente, usuário, gerente, desenvolvedor e testador. Podemos observar na figura 4, a representação de cada etapa da metodologia e os papéis envolvidos no processo.

Figura 4 – Representação dos papéis na metodologia YP



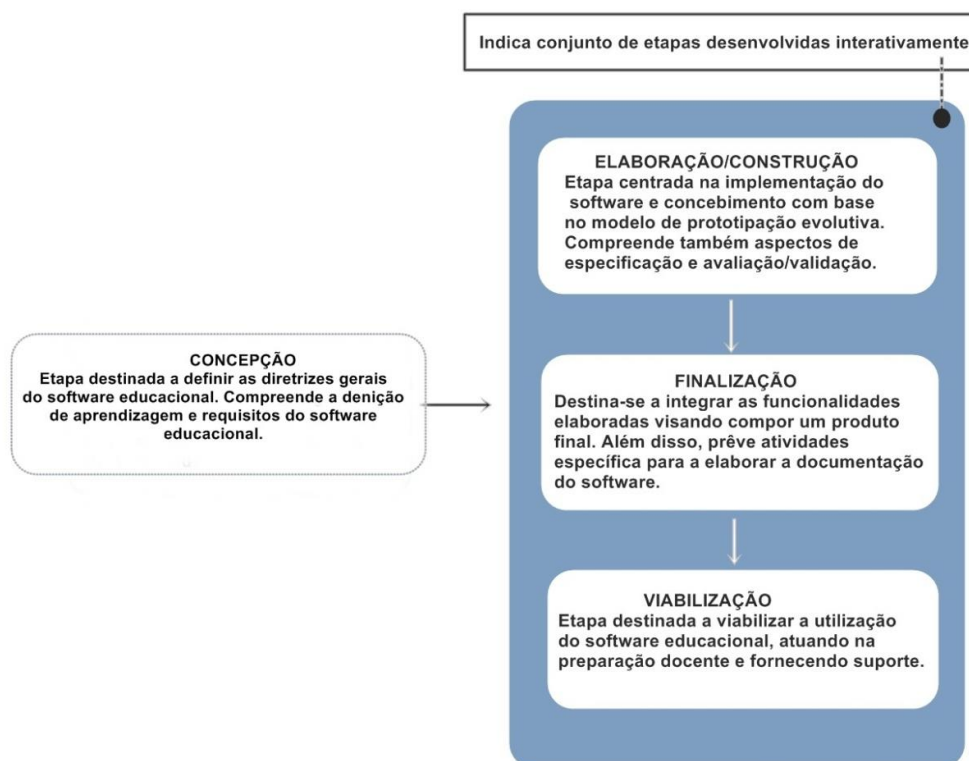
Fonte: Medeiros (2012, P.32).

Em resumo, o andamento dos processos na YP, segundo Garcia et al. (2004, p.05) “deve ser coordenado pelo gerente através da Reunião de Acompanhamento semanal que visa recolher e analisar métricas”.

2.5.3 METODOLOGIAS PARA O DESENVOLVIMENTO DE SOFTWARES EDUCACIONAIS

Poucos estudos foram elaborados pensando em formular uma metodologia para o desenvolvimento de softwares educacionais. No estudo realizado por Benitti et al. (2005) é apresentada uma proposta para o desenvolvimento de um software educacional. O processo de desenvolvimento é basicamente constituído de 4 (quatro) etapas principais: (i) concepção, (ii) elaboração, (iii) finalização e (iv) viabilização. Cada etapa possui um foco bem definido. A figura 5 descreve a característica de cada etapa.

Figura 5 – Visão Geral da metodologia proposta por Benitti et al. (2005)



Fonte: Autora, adaptada de Benitti et al. (2005)

Na etapa de concepção “cabe aos profissionais da educação conduzir a discussão em torno dos objetivos de aprendizagem que nortearam a concepção do software subsidiando a

definição dos requisitos computacionais, descritos pelos profissionais da área de computação (BENITTI et al. , 2005, p.3)”.

Na segunda fase, denominada de etapa de elaboração ou construção, são abrangidas as atividade de desenvolvimento, testes e validação.

Na etapa de finalização, “embora esta não seja a última etapa do processo de desenvolvimento, refere-se à parte final de “construção” do software, que somente ocorre após uma análise positiva da avaliação do uso do software educacional pelos alunos (BENITTI et al. , 2005, p.6)”.

A última fase é direcionada ao treinamento dos usuários em relação à utilização do software. Assim, “os profissionais da área computacional e de educação atuam em conjunto em todas as três atividades da etapa” (BENITTI ET AL, 2005, p.6). Nesta fase ocorre o treinamento dos docentes.

Há também o acompanhamento inicial do uso do software nessa etapa, pois todo software é sujeito a erros mesmo que as outras etapas tenham sido desenvolvidas corretamente.

Não foram encontradas outras metodologias para o desenvolvimento de softwares educacionais nos repositórios em que foi realizada a pesquisa. Mas, fica clara a necessidade de uma equipe pedagógica presente no desenvolvimento do software educacional.

2.5.4 TIPOS DE SOFTWARES EDUCACIONAIS

Com as pesquisas realizadas, verificou que para aplicar uma boa metodologia de software educacional é necessário identificar qual tipo de software será desenvolvido. Foram encontradas as seguintes classificações: software de exercícios; software de simulação; software aplicativos; software jogos educacionais.

Sobre os softwares de exercícios, Tavares e Silva (2017, p. 19) afirmam que, “esse tipo de software, além de apresentar o exercício, faz um apanhado das respostas, de modo a verificar o desempenho do usuário”.

Já os softwares simulações, “permitem ao aluno realizar atividades das quais normalmente não poderia participar, ou seja, por meio da simulação, é criada uma situação que se assemelha com a realidade (TAVARES; SILVA, 2017, p.20)”.

Softwares aplicativos são voltados para aplicações em atividades específicas como: processadores de texto e planilhas eletrônicas. Na educação, os jogos educacionais estimulam o desenvolvimento da imaginação da criança. Com eles, ela pode aprender e desenvolver suas habilidades de interação social.

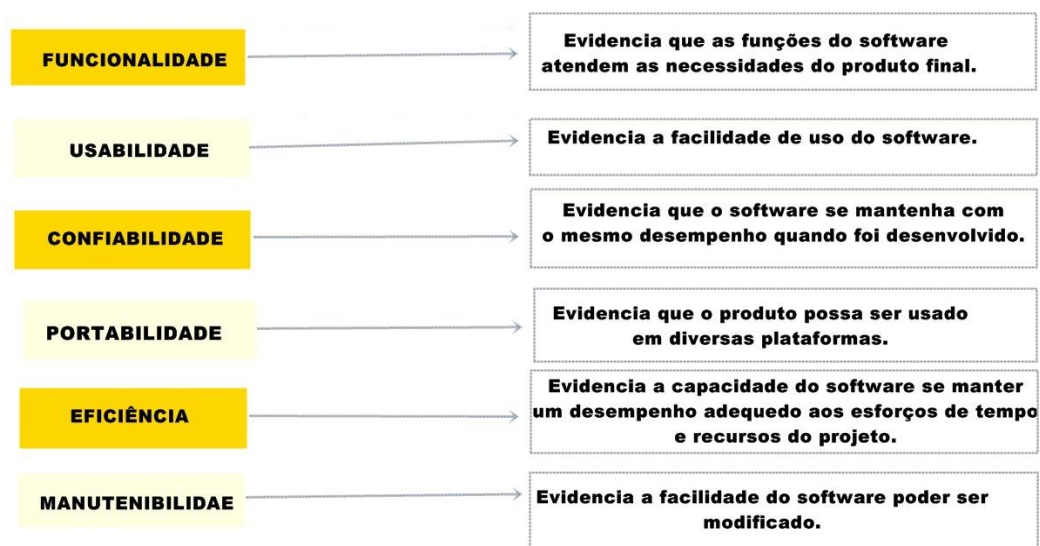
Os softwares do tipo jogos, geralmente são desenvolvidos para estimular o aprendizado,

criando um cenário competitivo entre o aluno e a máquina. “Os jogos educativos possibilitam ao aluno aprender de forma prazerosa e dinâmica, porque possuem desafios que despertam o interesse e a motivação no processo de ensino e aprendizagem (TAVARES; SILVA, 2017, p.5)”.

2.6 CARACTERÍSTICAS DE UM SOFTWARE PARA O ENSINO DA MATEMÁTICA

A Associação Brasileira de Normas Técnicas (ABNT), em 2003, estabeleceu NBR ISO/IEC 9126-1. Ela informa as características para avaliar a qualidade do produto de software. Essas características são: funcionalidade, usabilidade, confiabilidade, eficiência, manutenibilidade e portabilidade. A figura 6 descreve cada característica segundo esta norma.

Figura 6 – Características da qualidade de software segundo a NBR ISO/IEC 9126-1



Fonte: Autora, baseada na NBR ISO/IEC 9126-1, 2003.

Como as características de qualidade e as métricas associadas podem ser úteis não só à avaliação de produto de software, mas, também, para a definição de requisitos de qualidade e outros usos (NBR ISO/IEC 9126-1, 2003, p.2).

Mas a “Avaliação da Qualidade do Software Educacional” coloca em cena outros elementos, além das características anteriormente propostas pelas normas técnicas, pois a incorporação desses aplicativos só se justifica na medida em que possibilite um avanço qualitativo nos processos de ensino e aprendizagem, concorrendo para uma educação transformadora (GLADCHEFF et al., 2001, p.4).

Softwares matemáticos possuem outras características que devem ser levadas em

consideração além das citadas pela ISO/IEC 9126-1. Essas características como foi mencionada por Gladcheff et al. (2001), devem ser focadas em garantir o processo de ensino e aprendizado.

Os autores Lopes e Simião (2011, p.4) identificaram em seus estudos características que software matemático deve ter. Duas delas são:

- Documentação de descrição e manual do usuário, impresso ou online;
- Questões relacionadas a características pedagógicas gerais (objetivos, usabilidade, conteúdos matemáticos e praticidade).

Em complemento a essas características, no estudo realizado por Gladcheff et al., 2001, p.6-7) ele verifica características que devem ser seguidas em qualquer software educacional de Matemática do Ensino Fundamental. O quadro 1 mostra os aspectos técnicos que devem ser levados em consideração.

Quadro 1 – Aspectos técnicos que devem ser levados em consideração para um bom software matemático do ensino fundamental

DOCUMENTAÇÃO	SOFTWARE
Deve possuir instruções corretas de uso, instalação e desinstalação.	Fácil instalação e desinstalação;
Todas as atividades que o software realize devem ser descritas.	As funções disponíveis devem ser suficientes para realizar as tarefas pela qual o produto se propõe.
Não deve possuir erros gramaticais.	Requisitos de hardware e software compatíveis com os requisitos do computador onde será instalado o software.
Os termos usados na documentação devem estar no mesmo idioma que os usados na interface do software.	A empresa desenvolvedora deve fornecer suporte técnico e manutenção ao produto.

Fonte: Autora, 2019, baseada em Gladcheff et al., 2001, p.6-7)

Já em relação aos aspectos pedagógicos, Gladcheff et al.(2001), afirma que deve ser levado em consideração : O objetivo; A usabilidade; Os conceitos e a praticidade. Em relação aos objetivos, três pontos importantes devem ser levados em consideração. São eles:

1. Especificar os objetivos que se pretende alcançar em relação à Matemática, utilizando o produto como ferramenta de auxílio (GLADCHEFF et al. , 2001, p.7).
2. Se o software explora o conhecimento matemático dentro da realidade do aluno, a fim de que ele compreenda a Matemática como parte de sua vida cotidiana;

3. Expõem-se situações onde o aluno usa a linguagem Matemática para expressar-se com clareza e precisão;

Quanto a usabilidade, compreende-se com os estudos realizados por Gladcheff et al., 2001, a importância de verificar aspectos de interface. Deve ser analisado se, no software matemático, as funções são bem representadas; se possui sistema de ajuda; se a quantidade de informação de cada tela é apropriada para a faixa etária dos alunos e, se as animações são usadas de forma equilibrada.

Em relação aos conceitos, deve ser analisado o que se pretende trabalhar com os alunos através do software. É importante “refletir sobre a possibilidade de os conceitos matemáticos trabalhados pelo software serem relacionados com outros conceitos da Matemática e/ou de outras disciplinas (GLADCHEFF et al., 2001, p.8)”.

Em relação à praticidade, “caso julgue necessário, o professor deve verificar se o produto possui uma versão para ser utilizada em rede e se seu preço é compatível com o orçamento da escola (GLADCHEFF et al. , 2001, p.8)”.

2.7 COMO ESCOLHER UMA METODOLOGIA?

De acordo com Almeida (2017, p. 65-81) os fatores que compõem cenário para escolha do tipo de metodologia relacionada aos projetos de desenvolvimentos de softwares, devem levar em consideração as seguintes premissas:

- Duração do projeto. Projetos avaliados de acordo com seus prazos podem ser mais ou menos adequados em relação ao que se conhece sobre o uso dos diferentes tipos de metodologias de desenvolvimento de software;
- Complexidade do projeto. O grau de complexidade e o tipo de software ou sistema desenvolvido a cada projeto podem apresentar variações e melhores adequações a um tipo ou outro de metodologia de desenvolvimento;
- Quantidade de pessoas no projeto;
- Facilidade de comunicação e proximidade entre os membros da equipe;
- Viabilidade de comunicação e proximidade com o cliente ou área solicitante;
- Riscos e incerteza do projeto;
- Criticidade.

Para o desenvolvimento de software educacional, um fato que requer atenção da equipe desenvolvedora é que a metodologia escolhida deve possibilitar a avaliação do software antes da entrega, “avaliar um software educacional é uma etapa fundamental quando se visa alcançar

um ensino qualificado (FANTIN, 2017, p.3)”.

Assim, a escolha da metodologia de desenvolvimento do software ocorre quando a equipe é ciente dos prazos do projeto, da quantidade de pessoas envolvida e do grau de complexidade do projeto.

2.8 ETAPAS PARA ELABORAR E VALIDAR UMA METODOLOGIA DE SOFTWARE

Para entender os processos de uma metodologia de softwares, devemos ter em mente que o desenvolvimento de um software é uma atividade humana e que para garantir o sucesso do projeto é necessário usar ferramentas objetivas e úteis.

As metodologias ágeis devem ser baseadas em quatro valores: Interações e indivíduos são mais importantes que os processos e ferramentas; o software é mais importante que a documentação; o cliente estar presente durante o projeto e a metodologia deve apresentar agilidade nas mudanças.

As principais características das metodologias tradicionais são:

- Grande quantidade de documentos;
- “Clientes têm um papel importante durante o desenvolvimento da especificação, mas a participação dele é mínima em outras atividades (ALMEIDA, 2017, p.29)”.
- Divididas em etapas/fases. “Essas fases são muito bem definidas e englobam atividades como Análise, Modelagem, Desenvolvimento e Testes (Mainart e Santos, 2010, p.3)”.
- A conclusão de cada fase deve gerar um marco, que pode ser um documento, como Diagramas de UML, um protótipo ou versão do software (UTIDA, 2012, p.26).

As investidas para encontrar métodos de desenvolvimento de softwares educativos se mostra como um grande desafio. A falta de técnicas para assegurar a qualidade dos softwares educacionais, segundo Dalmon et al. (2012, p.9), acarretam problemas de manutenção e continuidade dos projetos, que em última instância acaba limitando seu uso em larga escala, mesmo que sua qualidade pedagógica seja alta”.

Compreende-se assim, que para elaborar uma metodologia é necessário investigar as características das metodologias existentes e o processo de validação ocorre de forma gradativa. É necessário utilizar a metodologia desenvolvida a fim de verificar sua eficiência.

3. RESULTADOS E DISCUSSÕES

Este capítulo apresentará os resultados obtidos com a pesquisa. No qual, será realizado um comparativo entre as principais metodologias presentes no mercado, a fim de buscar a que melhor contribuirá para a metodologia que será desenvolvida. Assim, esse capítulo possui duas seções. Na primeira, é investigada a metodologia de software que possa contribuir para a metodologia que será desenvolvida. Na última seção, é apresentada a metodologia de software desenvolvida a partir das pesquisas realizadas.

3.1 INVESTIGANDO CONTRIBUIÇÕES PARA O DESENVOLVIMENTO DA METODOLOGIA DE SOFTWARE

Com base nos estudos realizados notou-se que para o desenvolvimento de um bom software matemático em Libras, é necessário ter uma metodologia de software que atenda as seguintes características:

- Equipe multidisciplinar, formada por ouvintes e surdos;
- Matemáticos, pedagogos e interpretes de Libras, envolvidos em cada parte do processo;
- Testes de usabilidade em cada etapa desenvolvida;
- Definição do tipo de software e o conteúdo que será abordado;
- E por fim, levantamento de material didático sobre a temática a qual o software será destinado.

As características descritas acima devem ser respeitadas para garantir o sucesso do projeto. A fim de atribuir valor à metodologia que será desenvolvida na próxima seção, observou-se a necessidade de buscar uma metodologia de software, presente no mercado, que possa contribuir com alguns de seus métodos para a metodologia em desenvolvimento.

O quadro 2 a seguir, tem a finalidade de verificar uma metodologia que propicie a execução das fases que, no estudo de Benitti et al. (2005), são descritas.

Quadro 2 – Verificando uma metodologia de software que melhor se adéqua as fases propostas por Benitti et al. (2005)

Etapa	Descrição	Metodologias de software			
		YP	XP	Scrum	Cascata
Concepção	Equipe pedagógica para identificar as necessidades do software.				
Elaboração	Desenvolvimento, Teste e validação.	x	X	x	x
Finalização	Integração das partes do software.	x	x	x	x
Viabilização	Treinamento do usuário, equipe pedagógica e equipe de desenvolvimento atuando juntos no treinamento.				

Fonte: Autora, 2019

Como podemos observar na tabela 2, as metodologias só atendem a duas fases que Benitti et al. (2005) propôs em seu estudo, a fase de elaboração e finalização. Por ser dirigido a planos, o modelo em cascata será excluído, pois no atual cenário de desenvolvimento de software, é necessária a utilização de metodologias ágeis que proporcionam flexibilidade ao projeto.

As metodologias ágeis, YP, XP e Scrum, apesar de possuir características próprias, oferecem a empresa desenvolvedora um conjunto de ferramentas ágeis que facilitam o desenvolvimento do software. Com esse empate, notou-se a importância de utilizar recursos que as metodologias ágeis possuem. Assim, utilizaremos recursos das metodologias YP, XP e Scrum, para a elaboração da metodologia.

3.2 DESCRIÇÃO GERAL DA METODOLOGIA DE SOFTWARE MATEMÁTICO COM FOCO EM LIBRAS

Essa é uma versão piloto de uma metodologia de software matemático com foco em libras, pois, com os estudos realizados, encontrou-se inúmeros desafios para o desenvolvimento de um bom software educacional, principalmente no que se refere a softwares voltados a pessoas surdas.

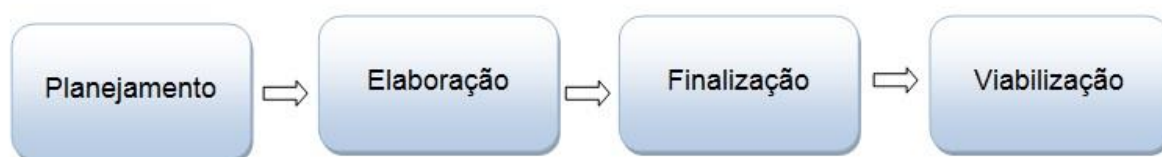
A matemática em si, é encarada como um grande desafio para a educação, principalmente para educação de surdos. Comprovando assim a necessidade de utilizar os recursos computacionais para aperfeiçoar o aprendizado em sala de aula pelos alunos surdos.

Esta metodologia terá como base fundamental, a presença de pessoas surdas em todo o processo de desenvolvimento do software. A equipe deve ser multidisciplinar, atendendo a 8 papéis. Que são eles: Interprete de libras, matemático, pedagogo, cliente, aluno, gerente, desenvolvedor e testador.

A fim de garantir o sucesso do projeto, o software desenvolvido deve seguir as diretrizes da base comum curricular brasileira do ensino de matemática nas fases iniciais.

Com base nos estudos realizados por Benitti et al. (2005) e nas metodologia ágeis explanada nesta pesquisa, optou-se que esta metodologia possua 4 etapas, como podemos observar na figura 5.

Figura 7 – Etapas da metodologia de software educacional matemático em libras



Fonte: Autora,2019

Assim como na metodologia YP, devem ser realizadas reuniões de acompanhamento do projeto. Estas reuniões devem ser orientadas pelo gerente e devem ocorrer ao final de cada tarefa concluída.

3.2 ETAPAS DA METODOLOGIA

a. PLANEJAMENTO

Esta é a primeira fase do processo. Ela é responsável por identificar o escopo do problema, realizar conversa com o cliente, informar os papéis envolvidos no processo, buscar estratégias educacionais, e criar dois documentos, assim como recomenda Benitti et al. (2005), o documento de planejamento geral do software educacional e o planejamento do processo.

✓ Identificação do escopo do problema

Baseada na metodologia YP, é a fase responsável por buscar conhecimentos sobre o problema que será solucionado. É nessa etapa que também é realizada uma pesquisa bibliográfica mais ampla sobre o conteúdo matemático que será abordado no software.

O parâmetro curricular nacional (PCN) de matemática deve ser a primeira fonte de pesquisa. O arquivo é disponibilizado no portal do MEC (portal.mec.gov.br) sendo o 3º volume da coletânea. Ao final dessa etapa, a equipe deve ser capaz de identificar os objetivos do software e o conteúdo que será abordado para realização à primeira conversa com o cliente.

✓ **Conversa com o cliente**

Baseada nas metodologias ágeis é realizada uma primeira conversa com o cliente, buscando informações sobre o sistema que será desenvolvido. Devem participar dessa conversa: o cliente, o gerente, desenvolvedor, testador, interprete de libras, matemático e pedagogo.

Antes da conversa devem ser elaborados o escopo do problema e um roteiro de perguntas para auxiliar o desenvolvedor a extrair o máximo de requisitos do sistema.

Durante a conversa o desenvolvedor deve extrair os requisitos funcionais e não funcionais do software, o perfil do usuário, os teste de aceitação, e outros.

O matemático e o pedagogo devem buscar os conteúdos que serão abordados no sistema solicitado pelo cliente. É nesse momento que o interprete de libras busca identificar os sinais matemáticos que serão necessários no software para traduzir o conteúdo para libras ou os melhores recursos gráficos para efetivar a transmissão de conteúdo.

Essa conversa, de acordo com a metodologia YP, deve durar 1 hora e meia. É nesta etapa que são geradas as diretrizes gerais do software educacional.

✓ **Definições de papéis**

Nesta metodologia deve-se optar por uma equipe multidisciplinar e que respeite acima de tudo o princípio da comunicação, assim como na metodologia Scrum, é importante que toda a equipe reconheça a importância do trabalho em equipe.

Recomenda-se para o desenvolvimento adequado do software a presença obrigatória de 8 papéis. A quantidade de cada um fica a critério do tamanho do projeto. Os papéis são: cliente, aluno, gerente, desenvolvedor, testador, interprete de libras, matemático e pedagogo.

Semelhante à metodologia Scrum, a equipe desenvolvedora também é multifuncional. Podem assumir diferentes papéis ao longo do projeto. Entretanto, é exigida a presença de interprete de libras, matemático e pedagogo, para autenticar o conteúdo do software.

Papéis

Cliente - É responsável por solicitar a equipe desenvolvedora o software e pode assumir o papel de usuário. Sua presença é ativa para o bom andamento do projeto. Assim como as metodologias XP, YP e Scrum o cliente deve ser capaz de transmitir o que ele deseja no software. Definir as características do software e o problema que ele espera solucionar.

O cliente deve informar qual o conteúdo que deve ser abordado no software, bem como qual etapa deve ser desenvolvido primeiro.

A cada tarefa concluída, o cliente deve estar presente para validar e confirmar se a funcionalidade cumpriu com suas exigências. Ele deve listar o que ele espera de usabilidade do software e como a linguagem de sinais deve ser abordada.

O cliente e o aluno devem trabalhar em conjunto para validar a interface e, só assim, a implementação do software será desenvolvida. Afinal, conforme mencionado na pesquisa bibliográfica, o cliente/ usuário entende que a interface é propriamente o software.

Aluno – É quem vai utilizar, de fato, o software. Sua presença é de suma importância no desenvolvimento do projeto. Pois o sistema deve ser desenvolvido pensando nele. O aluno deve ser surdo, suas responsabilidades são: validar a interface do software, avaliar o desempenho do software, verificar se a ferramenta está ensinando e gerando aprendizado a ele.

Gerente – É o responsável por orientar as atividades que serão desenvolvidas no projeto. Ele é responsável por conduzir as reuniões de acompanhamento, coletar as referências bibliográficas junto com o intérprete de libras, matemático e pedagogo.

Desenvolvedor – É responsável por modelar os requisitos do software para, posteriormente, utilizar uma linguagem de programação para produzir um código eficaz e correto. Ele deve participar da conversa com o cliente para levantar os requisitos funcionais e não funcionais do software. Sua presença é fundamental para auxiliar o gerente na escolha das tarefas que serão desenvolvidas.

Também é dever do desenvolvedor construir a lógica de dados do sistema, bem como sua interface gráfica. Ele também elabora o plano de teste do software e busca garantir a integração contínua do software.

Testador – É responsável por realizar revisões no código. Ele busca identificar erros no

software que os desenvolvedores não conseguem notar. Em softwares educacionais, seu papel é fundamental para garantir a qualidade das informações e a usabilidade do software.

Ao efetuar a revisão do código do testador pode realizar refatoração no código dos desenvolvedores para garantir códigos simples e claros. Ele também deve ser responsável por elaborar gerar os testes de aceitação.

Interprete de libras – O interprete de libras é o responsável por auxiliar os desenvolvedores na tradução da língua portuguesa para a Língua Brasileira de Sinais. Em parceria com o pedagogo, matemático e o gerente ele é responsável por levantar o conteúdo que será abordado no software. Sua presença é crucial em todo o desenvolvimento do projeto.

Alguns sinais de libras possuem duplo sentido ou são muito semelhantes e até mesmo, em alguns casos, não existe a tradução em se tratando da matemática. O que comprova a necessidade do intérprete em todas as etapas do projeto.

Pedagogo – É responsável por identificar as necessidades do aluno e traduzir para uma linguagem mais acessível os conceitos matemáticos, visando a efetividade do aprendizado. Na educação básica, ele é o responsável por inserir o aluno ao mundo acadêmico, se tornando um profissional multidisciplinar. No desenvolvimento de software deverá estar presente em todas as etapas, para garantir a efetivação do aprendizado através do software em desenvolvimento.

Matemático – Ele é responsável por garantir a eficiência dos conteúdos que serão abordados no software. Ele auxilia os testadores no desenvolvimento dos testes matemáticos a serem realizados para validar as funções desenvolvidas no sistema. Ele também estará presente em todas as etapas do projeto e deverá trabalhar em parceria direta com o pedagogo e o interprete de libras.

✓ **Estratégias educacionais**

A estratégia educacional consiste na análise das necessidades educacionais que o cliente especificou. É de responsabilidade do pedagogo, do intérprete de libras e do matemático buscarem estratégias para auxiliar a equipe no desenvolvimento da ferramenta educacional.

✓ **Planejamento geral do software educacional matemático com foco em libras**

Nesta etapa a equipe pedagógica (intérprete de libras, matemático e pedagogo) e a equipe desenvolvedora (gerente, desenvolvedor e testador) devem elaborar um documento que contenha:

- Visão geral do software;
- Características do problema;
- Objetivos de aprendizado;
- Requisitos funcionais e não funcionais do software.

Similar ao documento de visão da metodologia YP, esse documento não deve ser visto como o ponto final sobre o que deve ser feito. Os requisitos do software podem sofrer alterações durante todo o processo de desenvolvimento bem como os objetivos de aprendizado.

A visão geral do software deve ser, de forma objetiva e clara, apresentada o público alvo do software e suas características como usuários. Essas características devem ser identificadas na conversa com o cliente.

As características do problema e objetivos de aprendizado, inicialmente, serão identificadas na conversa com o cliente, mas posteriormente serão detalhadas na fase de estratégia educacional.

Os requisitos são os objetivos, funções e restrições que o software deve apresentar para solucionar o problema levantado pelo cliente. Os requisitos básicos devem ser descritos pelo cliente, entretanto, tendo em vista que está metodologia se trata de softwares educacionais matemáticos com foco em libras, os requisitos também devem ser especificados pela equipe pedagógica (o Matemático, Interprete de libras e o Pedagogo).

✓ **Planejamento do processo**

Esse documento é responsável por especificar o cronograma do projeto e identificar os requisitos e seus testes de aceitação.

O cronograma do projeto deve ser realizado por todos os membros da equipe. Com base nas metodologias Scrum e XP, o cronograma deve ser dividido em interações. Essas interações, baseadas na metodologia XP, devem ocorrer de forma cíclica, cujo tempo de cada uma deve variar de uma a três semanas e uma interação só deve ocorrer depois que a outra for concluída.

O cronograma pode mudar durante o decorrer do projeto, mas é sugerido que não

ocorram grandes mudanças para não comprometer o prazo final. As reuniões de acompanhamento devem ocorrer semanalmente, visando avaliar os resultados obtidos até o momento e validar a interação.

Similar à metodologia YP, a reunião deve ser orientada pelo gerente do projeto. Ao final de cada interação, deve ser apresentado ao cliente o que foi realizado e verificar se concorda com o que foi apresentado. Também, devem ser realizados testes com usuários surdos para identificar se a interação desenvolvida supre as necessidades do usuário.

Cada interação poderá cumprir vários requisitos, a decisão de quantos requisitos será abordada na interação fica a critério do gerente e das necessidades do cliente.

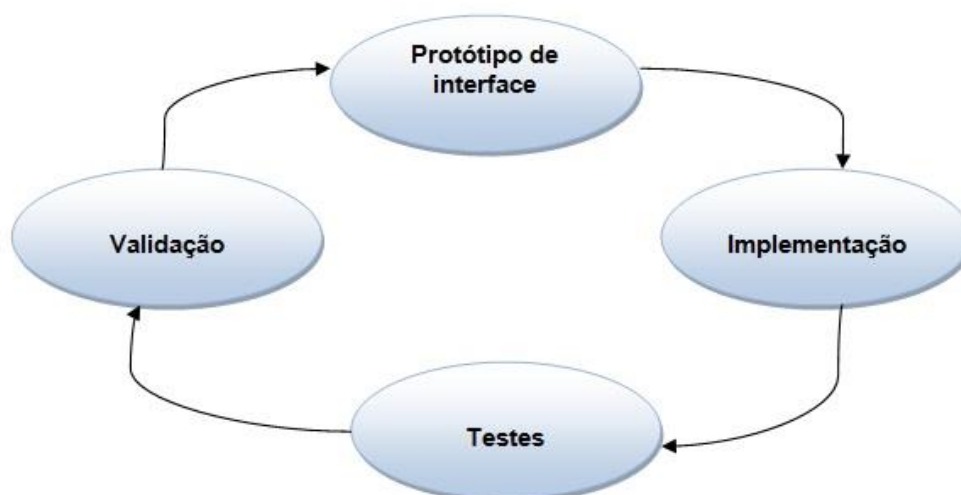
Cada requisito deve ter no mínimo um teste de usabilidade, que visa identificar se o objetivo educacional foi cumprido. Nesse teste, a equipe pedagógica deve estar envolvida para validar o objetivo educacional. Caso o requisito desenvolvido não cumpra as necessidades educacionais, a equipe pedagógica deve realizar novamente uma estratégia educacional para que a equipe desenvolvedora possa refatorar a interação. Essa refatoração acarreta mudanças no cronograma.

b. Elaboração

Etapa mais longa do processo, pois serão codificadas e testadas todas as etapas realizadas na fase de planejamento. É nela que são utilizados recursos computacionais para solucionar o problema do software. O primeiro passo que deve ser realizado é o projeto arquitetural.

As próximas fases (protótipo de interface, implementação, teste e validação) ocorreram de forma cíclica conforme a figura 6, pois como se trata de uma metodologia voltada para os anos iniciais do Ensino fundamental a interface gráfica deve ser atrativa para o aluno surdo. Dada como finalizado o ciclo é realizado a integração do software.

Figura 7 – Ciclo de elaboração



Fonte: Autora, 2019

O ciclo funciona assim: é criado o protótipo de interface, respeitando as diretrizes do software e as necessidades do usuário; implementa-se a interação; é realizado o teste para, só assim, validar e recomeçar o ciclo. Caso a validação não ocorra é inicializada, novamente, a interação.

✓ **Projeto arquitetural**

Primeira etapa da fase de elaboração. O desenvolvedor é livre para definir uma linguagem de programação e o banco de dados que cumpra com as necessidades do software. Essa definição deve ocorrer em uma reunião pequena, entre desenvolvedor e gerente, e tem duração média de 30 minutos.

Após esta definição é indicado que sejam realizados diagramas que possam ilustrar as atividades do sistema. Dentre esses diagramas que devem ser utilizados estão: o diagrama de classes, diagrama de caso de uso e diagrama de sequência.

A equipe é livre para decidir a quantidade de diagramas que eles irão utilizar, pois os mesmos devem ser explicados ao cliente para ele aprovar o projeto arquitetural.

A equipe pedagógica deve estar presente na elaboração do diagrama de caso de uso para que haja a efetivação do aprendizado. A forma armazenamento dos dados deve ser definida nessa etapa, em caso de softwares que necessitam de uma grande quantidade de informações, como é o caso dos softwares educacionais do tipo exercícios, a equipe pedagógica é fundamental para validar as formas de manipulação dos dados.

✓ **Protótipo de interface**

Dado por finalizado o projeto arquitetural, os desenvolvedores devem fazer protótipos de interface junto com a equipe pedagógica. Nessa etapa, é fundamental que toda a equipe se atente para a linguagem de sinais.

As estratégias educacionais escolhidas na etapa de planejamento são de suma importância e é a base para um bom desenvolvimento da interface. A interface deve ser desenvolvida para pessoas surdas e sua aprovação depende delas.

Além da validação do cliente, é necessária a validação do aluno surdo, pois, essa metodologia de desenvolvimento de software visa melhorar o aprendizado de pessoas surdas sobre matemática.

✓ **Implementação**

Fase em que são codificadas as interações descritas no planejamento do processo, o desenvolvedor deve estar atento a cada requisito funcional e não funcional que ele identificou na conversa com o cliente. A sua principal fonte de pesquisa deve ser os documentos disponíveis e a equipe pedagógica.

✓ **Testes**

Os testes devem ocorrer ao final de cada interação, o testador deve criar meios de avaliação e validação junto à equipe pedagógica. Mas, a principal fonte de validação do teste, deve ser do aluno. Nesse momento a equipe pedagógica observa e valida se está acontecendo à efetivação do aprendizado.

São exemplos de teste que devem ser realizados:

- Verificar se a interface é adequada para os alunos surdos. Devem ser levados em consideração, a linguagem, os artefatos visuais, a faixa etária do aluno e o nível de conhecimento do aluno em relação a libras;
- Verificar se o conteúdo de matemática está adequado para o aluno;
- Verificar se todas as funções, na qual o software foi desenvolvido, se comportam de maneira correta;
- Verificar o aprendizado do aluno ao utilizar a ferramenta.

Assim, os exemplos aqui dados, devem ser um guia para a elaboração de outros testes. Devem existir testes realizados pelo testador, outros realizados pela equipe pedagógica e testes realizados pelos alunos surdos.

✓ **Validação**

O processo de validação ocorre de dois lados. Primeiramente deve ocorrer com a equipe de desenvolvimento e equipe pedagógica. Após essa validação, o cliente e os alunos devem validar a interação, onde o gerente do projeto, junto da equipe pedagógica e desenvolvedora, apresenta ao cliente e ao aluno o resultado da interação desenvolvida, bem como os testes realizados, a fim de verificar a eficiência do aprendizado do aluno.

✓ **Integração do software**

Nesta etapa os desenvolvedores devem unir todas as interações feitas. Ao final dessa integração, os testadores realizam o teste usando todas as funções do software junto da equipe pedagógica e, em seguida, com os alunos. Ao término da integração e do teste final a equipe pedagógica deve avaliar se o software atende as necessidades educacionais a ser solucionados para, só assim, o cliente aprovar o software.

c. Finalização

Conforme Benitti et al. (2005), essa etapa consiste em realizar duas etapas: a **integração do sistema**, que consiste na união de todos os artefatos desenvolvidos na etapa de elaboração; e a **documentação**, que deve ser um manual para o aluno e para os professores.

Nessa documentação deve ser deixado claro as funções do software e os objetivos educacionais. É obrigatório também, que esse documento seja adaptado em vídeo utilizando a linguagem de sinais. O tutorial de uso deve estar disponível no software e a comunicação deve ocorrer em libras. Ao final dessa etapa, o software já pode ser utilizado pelos alunos surdos.

d. Viabilização

Última etapa do desenvolvimento do software, seguindo as especificações de Benitti et al. (2005) e dos estudos realizados, é nessa etapa que deve ser realizado treinamentos com a

equipe docente e discente. Nessa etapa o interprete de libras, pedagogo, matemático e gerente devem realizar treinamentos com os docentes que irão utilizar a ferramenta. A empresa, mesmo que não desenvolva o software para uma instituição de ensino, deve disponibilizar um treinamento direcionado aos docentes.

Diferente dos softwares em geral, os softwares voltados à educação necessitam de um apoio maior da equipe desenvolvedora, por isso é necessário que todo software desenvolvido tenha a documentação e a adaptação em vídeo, utilizando a linguagem de sinais.

O treinamento com aluno pode ocorrer de duas formas: **presencial** com a presença do interprete de libras, pedagogo, matemático e gerente ou com um **tutorial** em vídeo usando libras para realizar o treinamento.

4. CONCLUSÕES E CONSIDERAÇÕES FINAIS

Diante da problemática levantada, a atual pesquisa se mostrou desafiadora visto que, as metodologias de desenvolvimento de software educacionais ainda é um grande desafio para a engenharia de software.

Utilizando as *strings* “Educação de surdos”, “alunos e professores”, “matemática para o ensino fundamental”, “educação e inclusão”, “Linguagem brasileira de sinais(libras)”, “praticas pedagógicas para o ensino de matemática para alunos com deficiência auditiva”, “tecnologias utilizadas em sala de aula para o ensino da matemática para alunos do 1º ao 5º”, “aplicativos para o ensino de matemática em libras”, foi entrada aproximadamente 14 mil artigos relacionados.

Apesar da quantidade de artigos, poucos apresentaram contribuições para a presente pesquisa. Apenas 50 deles foram lidos. E usando as *strings*, “Metodologias para desenvolvimento de softwares educacionais”, “metodologias de desenvolvimento de softwares”, “metodologias de desenvolvimento de softwares educacionais de matemática”, só três artigos falavam de metodologias de desenvolvimento de softwares voltados para a educação e não foi encontrado nenhum falando sobre metodologia de desenvolvimento de software educacional para o ensino de matemática a surdos.

Foi encontrado poucos estudos relacionados ao uso de softwares educacionais para pessoas surdas. Se tratando do objetivo específico de verificar os estudos existentes sobre softwares matemáticos para o ensino a pessoas surdas não foi satisfatório apenas foi encontrado um estudo sobre o aplicativo matematicando.

As investigações realizadas a fim de verificar as metodologias educacionais comprovaram a necessidade de se discutir a criação de novas técnicas de desenvolvimento de software focado na educação.

Ao investigar a dificuldade do ensino da matemática para alunos surdos, verificou a necessidade de se realizar estudos sobre essa temática. A matemática ainda é um grande desafio para os alunos surdos. Muitos sinais matemáticos em libras são parecidos e requer uma atenção maior dos educadores ao transmitirem os conteúdos.

A educação de pessoas surdas é desafiadora para o meio acadêmico, principalmente se tratando do ensino da matemática. Autores como Bertoli (2012) apresentam a adaptação curricular da matemática como um grande desafio para a educação de surdos.

Pode-se notar que para um bom desenvolvimento de software educacional necessita-se de uma equipe pedagógica envolvida em todo o processo, e, se tratando de softwares voltados

a pessoas surdas, é necessária a presença de interpretes de libras e alunos surdos em todo o processo. Assim, o diferencial da metodologia aqui proposta é a presença de uma equipe pedagógica e de alunos surdos em todo o processo de desenvolvimento do software.

4.1 LIMITAÇÕES

Por se tratar de uma monografia, a metodologia foi desenvolvida sobre o olhar da autora. Mas, é necessária a participação de várias pessoas, que possam assumir diferentes papéis e que realizem testes a fim de validar a metodologia.

4.2 TRABALHOS FUTUROS

A metodologia aqui desenvolvida necessita que sejam realizados testes para verificar a sua eficiência. Esta é uma versão piloto, que visa contribuir para trabalhos futuros relacionados ao desenvolvimento de software educacional matemáticos com foco em libras.

Vale salientar que, apesar da metodologia se limitar ao desenvolvimento dos softwares matemáticas para os anos iniciais do ensino fundamental, é possível realizar adaptações e selecionar novos papeis e atribuições à metodologia.

REFERÊNCIAS

ALMEIDA, Guilherme Augusto Machado de. **Fatores de escolha entre metodologias de desenvolvimento de software tradicionais e ágeis**. 2017. Disponível em: <<https://www.teses.usp.br/teses/disponiveis/3/3136/tde-11042017-143311/publico/GuilhermeAugustoMachadodeAlmeidaCorr17.pdf>>. Acessado em: 27 de abril de 2019.

ALVES, Luana Leal. **A Importância Da Matemática Nos Anos Iniciais**. 2016. Disponível em: <<https://wp.ufpel.edu.br/geemai/files/2017/11/A-IMPORT%C3%82NCIA-DAMATEM%C3%81TICA-NOS-ANOS-INICIAS.pdf>>. Acessado em: 14 de abril de 2019.

ARAÚJO, Enio Gomes. **Ensino de matemática em libras: reflexões sobre minha experiência numa escola especializada**. / Enio Gomes Araujo. 2016. Disponível em: <<http://www.matematicainclusiva.net.br/pdf/TESEENIOFINAL.pdf>>. Acessado em: 20 de Setembro de 2018.

Associação Brasileira de Normas Técnicas (ABNT). **NBR ISO/IEC 9126-1: Engenharia de software - Qualidade de produto Parte 1: Modelo de qualidade**. 2003. Disponível em: <https://aplicacoes.mds.gov.br/sagirms/simulacao/sum_executivo/pdf/fichatecnica_21.pdf>. Acessada em: 16 de maio 2019.

BASSANI, Patricia Scherer; PASSERINO, Liliana M.; PASQUALOTTI, Paulo R.; Ritzel, Marcelo Iserhardt. **Em busca de uma proposta metodológica para o desenvolvimento de software educativo colaborativo**. 2006. Disponível em: <http://www.cinted.ufrgs.br/renoteold/jul2006/artigosrenote/a30_21191.pdf>. Acessado em: 10 de maio de 2019.

BENITTI, Fabiane Barreto Vavassori; SEARA, Everton Flávio Rufino; SCHLINDWEIN, Luciane Maria. **Processo de Desenvolvimento de Software Educacional: proposta e experimentação**. 2005. Disponível em: <<https://seer.ufrgs.br/renote/article/viewFile/13849/8025>>. Acessado em: 1 de novembro de 2018.

BERTOLI, Vaneila. **O Ensino Da Matemática Para Alunos Surdos**. 2012. Disponível em: <<http://www.sinect.com.br/anais2012/html/artigos/ensino%20mat/34.pdf>>. Acessado em: 28 de Setembro de 2018.

BRASIL. **Secretaria de Educação Fundamental. Parâmetros curriculares nacionais : matemática / Secretaria de Educação Fundamental**. – Brasília : MEC/SEF, 1997. 142p. 1. Parâmetros curriculares nacionais. 2. Matemática : Ensino de primeira à quarta série. I. Título. Disponível em: <<http://portal.mec.gov.br/seb/arquivos/pdf/livro03.pdf>>. Acessado em: 10 de maio de 2019.

BRASIL. **LEI Nº 10.436, DE 24 DE ABRIL DE 2002**. Dispõe sobre a Língua Brasileira de Sinais - Libras e dá outras providências, Brasília, DF, abril de 2002.

BRASIL. **LEI Nº 13.146, DE 6 DE JULHO DE 2015**. Institui a Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência). Brasília, DF, julho 2015.

CALIXTO, Renato Messias Ferreira; GARCÊZ, Regiane Lucas de Oliveira; OLIVEIRA, Sônia Marta de. **Traduzir e interpretar incursões no mundo do outro ou atos de fronteira? Reflexões teóricas sobre o papel do intérprete a partir de uma perspectiva culturalista.** 2012. Disponível em:

<http://www.congressotils.com.br/anais/anais/tils2012_traducao_questao_calixtogarcez.pdf>. Acessado em: 30 de Setembro de 2018.

CARNEIRO, Luís Miguel Nunes. **A importância de Métodos inovadores de comunicação na escola.** 2013. Disponível em:

<<http://repositorio.uportu.pt/xmlui/bitstream/handle/11328/670/TME%20507.pdf?sequence=1&isAllowed=y>>. Acessado em: 20 de março de 2019.

CHAMMAS, Adriana; QUARESMA, Manuela; MONT'ALVÃO, Cláudia. **Um enfoque ergonômico sobre a metodologia de design de interfaces digitais para dispositivos móveis.** 2013. Disponível em:

<<https://www.epublicacoes.uerj.br/index.php/arcosdesign/article/viewFile/12188/9547>>. Acessado em: 03 de maio de 2019.

CHAVES, Sandra Wirgynnia de Figueiredo. **Desenvolvimento e avaliação de um software educativo como apoio no processo de ensino-aprendizagem da saúde da criança.** 2017. Dissertação (mestrado profissional) – Universidade Estadual do Ceará, Centro de ciências da saúde, mestrado profissional em ensino na saúde, Fortaleza.

COSTA, Maria Stela Oliveira. **Os benefícios da informática na educação dos surdos.** 2011. Disponível em: <<https://periodicos.furg.br/momento/article/view/2271>>. Acessado em: 03 de maio de 2019.

DALMON, Danilo L.; BRANDÃO, Anarosa A. F.; O. BRANDÃO, Leônidas. **Uso de Métodos e Técnicas para Desenvolvimento de Software Educacional em Universidades Brasileiras.** 2012.

DOURADO, Gabriel Costa; OLIVEIRA, Genaina Sousa de; SANTOS, Neuzilene Soares dos; SILVA, Ivanete Cirqueira e; COSTA, Walber Christiano Lima da. **Aplicativo Matematicando: O Ensino Das Quatro Operações Para Alunos Surdos.** 2017. Disponível em:

<https://editorarealize.com.br/revistas/conedu/trabalhos/TRABALHO_EV073_MD4_SA10_ID5646_11092017171337.pdf>. Acessado em: 16 de setembro de 2018.

F. G. Silva; S. C. P. Hoentsch, L. Silva. **Uma análise das Metodologias Ágeis FDD e Scrum sob a Perspectiva do Modelo de Qualidade MPS.BR. 2009.** Disponível em: <<https://www.scienciaplena.org.br/sp/article/view/678/348>>. Acessado em: 27 de abril de 2019.

FANTIN, Kátia. **Metodologia de Avaliação de Software Educacional.** 2017. Disponível em: <

<https://repositorio.ucs.br/xmlui/bitstream/handle/11338/3080/TCC%20Katia%20Fantin.pdf?sequence=1&isAllowed=y>>. Acessado em: 4 de abril de 2019.

FERNANDE, Roberto Fabiano; FERENHOF, Helio A.; MIGUEZ, Viviane B.; TEZA, Pierry; SOUZA, João Artur de; ABREU, Aline França de; DANDOLINI, Gertrudes Aparecida. **A agilidade do framework scrum como uma prática viral de disseminação do**

conhecimento. 2012. Disponível em:

<<http://www.ijopm.org/index.php/IJOPM/article/view/87>>. Acessado em: 15 de abril de 2019.

FREITAS, Victor Amaral; MEDEIROS, Sheyla Natália de; SOUSA, Hercilio de Medeiros; **Importância do software Vibras no processo de aprendizagem de pessoas com deficiências auditivas**. 2017. Disponível em: <<http://www.tise.cl/volumen13/TISE2017/81.pdf>>. Acessado em: 29 de Setembro de 2018.

GAMA, Flausino José da. **Iconographia dos signaes dos surdos-mudos / Flausino José da Gama**. — Rio de Janeiro: INES, 2011. (Série Histórica do Instituto Nacional de Educação de Surdos; 1). Disponível em: <<https://drive.google.com/file/d/0B6WyKJSZvdJdb0M2RjhqcFVBOG8/view>>. Acessado em: 28 de agosto de 2018.

GARCIA, Francilene Procópio ;LIMA, Aliandro Higino Guedes;FERREIRA, Danilo de Sousa , JUNIOR, Fábio Luiz Leite;ROCHA, Giselle Regina Chaves da; MENDES, Gustavo Wagner Diniz;PONTES, Renata França de; ROCHA, Verlaynne Kelley da Hora Rocha;DANTA, Vinicius Farias. **easYProcess: Um Processo de Desenvolvimento para Uso no Ambiente Acadêmico**. 2004. Disponível em: <<http://gustavowagner.com/paginaPessoal/documentos/artigos/YPsbc2004-GustavoWagnerMendes.pdf>>. Acessado em: **23 de março de 2019**.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 4.ed. São Paulo: Atlas, 2002

GLADCHEFF, Ana Paula; ZUFFI, Edna Maura; SILVA, Dilma Menezes da. **Um Instrumento para Avaliação da Qualidade de Softwares Educacionais de Matemática para o Ensino Fundamental**. 2001. Disponível em: <http://www.pucrs.br/ciencias/viali/tic_literatura/artigos/pacotes/Avalia%C3%A7%C3%A3o%20de%20software%20educativo%20para%20o%20ensino%20da%20matem%C3%A1tica%20do%20fundamental.pdf>. Acessado em: 05 de abril de 2019.

LOBATO, Maria José Silva. NORONHA, Claudianny Amorim. **O aluno surdo e o ensino de matemática: desafios e perspectivas na escola regular de ensino em Natal, RN**. 2013. Disponível em: <<http://www.conferencias.ulbra.br/index.php/ciem/vi/paper/viewFile/815/69>>. Acessado em: 27 de abril de 2019.

LORIGGIO , Alexandre Fekettia; FARIAS , Victor Mitsunaga; MUSTARO, Pollyana Notargiacomo. **Aplicações de gamificação e técnicas de motivação à aprendizagem da metodologia ágil scrum**. 2013. Disponível em: <<http://copec.eu/congresses/icece2013/proc/works/73.pdf> >. Acessado em: 20 de abril de 2019.

MACHADO, Thiago Ganimi. **Metodologia Ágil – Extreme Programming: Programação Em Par**. 2009. Disponível em: <<https://cepein.femanet.com.br/BDigital/arqTccs/0511150121.pdf>>. Acessado em 14 de abril de 2019.

MAINART, Domingos de A; SANTOS, Ciro M. **Desenvolvimento de Software: Processos**

Ágeis ou Tradicionais? Uma visão crítica. 2010. Disponível em: <http://www.enacomp.com.br/2010/anais/artigos/completos/enacomp2010_4.pdf>. Acessado em: 27 de abril de 2019.

MARÇAL, Edgar; LIMA, Luciana de; JUNIOR, Melo; VIANA, Windson; ANDRADE, Rossana; RIBEIRO, Júlio Wilson Ribeiro. **Da Elicitação de Requisitos ao Desenvolvimento de Aplicações de Mobile Learning em Matemática.** Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/sbie/2010/0028.pdf>>. Acessado em: 01 de maio de 2019.

MEDEIROS, A. F. **Elicitação de critérios essenciais para a adaptação de uma metodologia ágil para o desenvolvimento de software educativo.** 2012. 82 f. Trabalho de Conclusão de Curso (Graduação em Computação)—Centro de Ciências Exatas e Sociais Aplicadas, Universidade Estadual da Paraíba, Patos, 2012. Disponível em: <<http://dspace.bc.uepb.edu.br/jspui/handle/123456789/3689>>. Acessado em: 01 de maio de 2019.

MORI, Nerli Nonato Ribeiro e SANDER, Ricardo Ernani. **HISTÓRIA DA EDUCAÇÃO DOS SURDOS NO BRASIL. 2015.** Disponível em: <http://www.ppe.uem.br/publicacoes/seminario_ppe_2015/trabalhos/co_04/94.pdf>. Acessado em: 5 de setembro de 2018.

Nunes, Rodrigo Dantas. **A implantação das metodologias ágeis de desenvolvimento de software scrum e extreme programming (xp): uma alternativa para pequenas empresas do setor de tecnologia da informação.** 2016. Disponível em: <<http://www.forscience.ifmg.edu.br/forscience/index.php/forscience/article/view/117/134>>. Acessado em: 8 de abril de 2019.

Organização Internacional para Padronização. **ISO 9241-210: Ergonomia da interação humano-sistema - Parte 210: Design centrado no ser humano para sistemas interativos.** 2010. Disponível em: <<https://www.sis.se/api/document/preview/912053/>>. Acessado em: 15 de maio de 2019.

REIS, Luana Silva; SALES, Angelina S. da Silva; LIMA, Maria Dayane F. Cirino; ARAÚJO, Tiago Maritan U. de. **Avaliação de Usabilidade do Aplicativo VLibras-Móvel com Usuários Surdos.** Disponível em: <<http://www.inf.ufrgs.br/webmedia2017/wpcontent/anaiswebmedia/files/wtic/wtic10.pdf>>. Acessado em: 28 de Setembro de 2018.

ROCHA, S. **O INES e a educação de surdos no Brasil: aspectos da trajetória do Instituto Nacional de Educação de Surdos em seu percurso de 150 anos.** Rio de Janeiro: INES, 2008.

SANTOS, Elton Andrade Dos. **Escolha E Implantação De Uma Metodologia: Um Estudo De Caso Para O Laboratório De Aplicação Em Tecnologia Da Informação.** 2007. Disponível em: <https://projetos.inf.ufsc.br/arquivos_projetos/projeto_722/Artigo.pdf>. Acessado em: 05 de abril de 2019.

SILVA, IramiBila da; VASCONCELOS, Carlos Alberto. **Linguagem matemática em libras: comunicação e ensino.** 2015. Disponível em:

<<https://eventos.set.edu.br/index.php/enfope/article/viewFile/1196/296>>. Acessado em: 21 de Setembro de 2018.

SILVA, Noêmia Barbosa da; GUIMARÃES, Maria Ivone Pereira. **O Uso Da Informática No Processo De Ensino/Aprendizagem De Alunos/As Com Necessidades Educacionais Especiais (Deficiente Auditivo)**. 2014. Disponível em: <https://semanaacademica.org.br/system/files/artigos/artigo_para_publicar_3.pdf>. Acessado em: 29 de outubro de 2018.

SOARES, Michel dos Santos. **Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software**. 2004. Disponível em: <<http://www.dcc.ufla.br/infocomp/index.php/INFOCOMP/article/view/68>>. Acessado em: 08 de abril de 2019.

SOMMERVILLE, Ian. **Engenharia de Software** / Ian Sommerville ; tradução Ivan Bosnic e Kalinka G. de O. Gonçalves; revisão técnica KechiHirama. — 9. ed. — São Paulo : Pearson Prentice Hall, 2011.

SOUZA, Luciano Malaquias de. **MÉTODO ÁGIL XP (EXTREME PROGRAMMING)**. 2007. Disponível em: <http://intranet.fainam.edu.br/aceso_site/fia/academos/revista3/6.pdf>. Acessado em: 14 de abril de 2019.

SOUZA, Renato Máximo de 1 SILVA, Regina Célia Marques Freitas. **A Interface Com O Usuário No Projeto De Software**. 2009. Disponível em: <<http://www.nucleus.feituverava.com.br/index.php/nucleus/article/view/148>>. Acessado em: 8 de setembro de 2018.

STROBEL, Karin Lílian. **A VISÃO ESCOLAS HISTÓRICA DA IN(EX)CLUSÃO DOS SURDOS NAS ESCOLAS**. 2006. Disponível em: <<https://periodicos.sbu.unicamp.br/ojs/index.php/etd/article/view/806/821>>. Acessado em: 8 de setembro de 2018.

STUMPF, Marianne Rossi. **Educação de Surdos e Novas Tecnologias**. 2010. Disponível em: <http://www.libras.ufsc.br/colecaoLetrasLibras/eixoFormacaoPedagogico/educacaoDeSurdosENovasTecnologias/assets/719/TextoEduTecnologia1_Texto_base_Atualizado_1_.pdf>. Acessado em: 1 de outubro de 2018.

TAVARES, Jéssika Lima. **Modelos, Técnicas E Instrumentos De Análise De Softwares Educacionais**. 2017. Disponível em: <<https://repositorio.ufpb.br/jspui/bitstream/123456789/2563/1/JLT19062017.pdf>>. acessado em: 20 de abril de 2019.

TAVARES, Jéssika Lima; SILVA, Lebiam Tamar Gomes. **TIPOS E CLASSIFICAÇÕES DE SOFTWARES EDUCACIONAIS**. 2017. Disponível em: <https://www.editorarealize.com.br/revistas/conedu/trabalhos/TRABALHO_EV073_MD1_SA19_ID4841_29082017144046.pdf>, Acessado em: 23 de março de 2019.

UTIDA, Kleber Hiroki. **Metodologias tradicionais e metodologias ágeis: análise comparativa entre ratioml Unified process e extreme programming**. 2012. Disponível em: <<http://www.fatecsp.br/dti/tcc/tcc00055.pdf>> .Acessado em: 23 de março de 2019.