



## ESTUDO SOBRE REQUISITOS ESSENCIAIS NA ANÁLISE DE SISTEMAS PARA TECNOLOGIA ASSISTIVA

GIRLAINE DAMÁZIO DA SILVA<sup>1</sup>; LUDYMYLLA MARIA DE OLIVEIRA<sup>2</sup>;  
TALLISON WANDERLEY AMORIM FERREIRA<sup>3</sup>; NILTON FREITAS JUNIOR<sup>4</sup>

<sup>1</sup>Bacharel em Sistemas de Informação. Universidade do Estado de Minas Gerais – Unidade Carangola. E-mail: [girlaine.1291649@discente.uemg.br](mailto:girlaine.1291649@discente.uemg.br).

<sup>2</sup>Bacharel em Sistemas de Informação. Universidade do Estado de Minas Gerais – Unidade Carangola. E-mail: [ludymylla.1291687@discente.uemg.br](mailto:ludymylla.1291687@discente.uemg.br).

<sup>3</sup>Graduando em Sistemas de Informação. Universidade do Estado de Minas Gerais – Unidade Carangola. E-mail: [tallison.1234235@discente.uemg.br](mailto:tallison.1234235@discente.uemg.br).

<sup>4</sup>Mestre em Pesquisa Operacional e Inteligência Computacional. Universidade do Estado de Minas Gerais – Unidade Carangola. E-mail: [niltonfjunior@gmail.com](mailto:niltonfjunior@gmail.com).

### RESUMO

A Tecnologia da Informação mostra a força de sua presença em vários contextos da sociedade contemporânea, tornando-se fator relevante não só no gerenciamento de informações de ambientes empresariais ou nas tarefas cotidianas, mas também como agente transformador da inclusão de todos os indivíduos aos inúmeros benefícios que seus recursos proporcionam. A construção de uma sociedade inclusiva também envolve igualdade de oportunidades na utilização de recursos de Tecnologia da Informação, tornando deveras importante que se desenvolvam estudos relacionados à Tecnologia Assistiva, cujos esforços estabelecem o intuito de ampliar as habilidades funcionais de pessoas com deficiência, proporcionando uma maior inclusão e independência de vida. O presente trabalho apresenta um estudo sobre requisitos essenciais na Análise de Sistemas para aplicação específica em projetos de *softwares* que procurem contemplar aspectos da Tecnologia Assistiva. Foram utilizados procedimentos de levantamento de requisitos com padrões de trabalho de Metodologia Ágil SCRUM, com a participação de um *Product Owner* que detém propriedade de conhecimento sobre o tema. A partir desse levantamento, o trabalho apresenta sua proposta de incluir, no ciclo de vida do SCRUM, um novo artefato de análise, chamado aqui de *Special Product Backlog*, cujo conceito é a inclusão de requisitos especificamente assistivos na criação de um software. Desta maneira, foi possível apresentar um modelo de adição de requisitos que podem ser considerados essenciais para análise e desenvolvimentos de softwares com aspectos de Tecnologia Assistiva, mantendo este trabalho com foco para pessoas com deficiência visual, mas orientando-se também na premissa de sua aplicação em outros tipos de requisitos que atendam à desenvolvimentos voltados às necessidades de outros tipos de deficiência.

**Palavras-chave:** Análise de Requisitos; Engenharia de Software; Tecnologia Assistiva.

### STUDY ON ESSENTIAL REQUIREMENTS IN THE ANALYSIS OF SYSTEMS FOR ASSISTIVE TECHNOLOGY

### ABSTRACT

Information Technology shows the strength of its presence in various contexts of contemporary society, becoming a relevant factor not only in managing information in business environments or in everyday tasks, but also as a transforming agent for the inclusion of all individuals to the countless benefits that its resources provide. The construction of an inclusive society also involves equal opportunities in the use of Information Technology resources, making it very important to develop studies related to Assistive Technology, whose efforts aim to expand the functional skills of people with disabilities, providing a greater inclusion and independence of life. The present work presents a study on essential requirements in Systems Analysis for specific application in software projects that seek to contemplate aspects of Assistive Technology. Requirements gathering procedures were used with SCRUM Agile Methodology work standards, with the participation of a Product Owner who owns knowledge on the subject. From this survey, the work presents its proposal to include, in the SCRUM

---

life cycle, a new analysis artifact, called here Special Product Backlog, whose concept is the inclusion of specifically assistive requirements in the creation of a software. In this way, it was possible to present a model for adding requirements that can be considered essential for the analysis and development of software with aspects of Assistive Technology, keeping this work focused on people with visual impairments, but also guided by the premise of its application in other types of requirements that meet developments aimed at the needs of other types of disabilities.

**Keywords:** Requirements Analysis; Software Engineering; Assistive Technology.

## INTRODUÇÃO

Toda evolução observada para a Tecnologia da Informação (TI) mostra que a aplicação de seus recursos tem ampliado continuamente, indo muito além dos trabalhos de processamentos de dados, estando presente hoje nos mais diversos contextos da sociedade moderna. Desde usos em tarefas domésticas até o apoio ao gerenciamento completo de organizações, a TI se tornou um dos principais meios de acesso a informações e disseminação de conhecimentos (VELOSO, 2012).

Considerando as oportunidades que a TI proporciona para acesso a conhecimentos diversos, tem-se no âmbito da Educação a possibilidade de benefícios no processo de ensino e aprendizagem. Isso ocorre através de recursos de *software* que podem produzir inovações em técnicas e habilidades educacionais. Há, portanto, a possibilidade de cenários favoráveis ao aperfeiçoamento de competências e habilidades pessoais e profissionais por meio do constante acesso à informação (CAMPOS; BLIKSTEIN, 2019).

Toda esta abrangência, traz consigo a necessidade de apreciações particulares para as aplicabilidades da TI. A versatilidade da tecnologia permite que o acesso à informação, em especial para processos de ensino e aprendizado, esteja ao alcance de todos, sem obstáculos que impeçam inclusões para sua utilização. Tem-se, então, a aproximação da TI para com as temáticas de acessibilidade e o conceito de Tecnologia Assistiva (MIZRAHI; CICERO, 2016).

Embora o termo Tecnologia Assistiva seja considerado recente, ele identifica o conjunto de recursos e serviços relacionados à TI, que visam contribuir na ampliação das habilidades funcionais de pessoas com deficiência, buscando proporcionar maior inclusão e independência de vida (BERSCH; TONOLLI, 2006 *apud* BERSCH 2017).

Considera-se que a construção de uma sociedade inclusiva, com plena participação e igualdade na tecnologia, tem como um de seus princípios a interação efetiva de todos os cidadãos, com ênfase no conceito de inclusão digital. Nesta perspectiva, é importante que se desenvolvam estudos relacionados à esta temática, que possam contribuir para a produção de conhecimentos sobre ações de Tecnologia Assistiva.

Entretanto, ao se considerar o desenvolvimento de *softwares* voltados para aplicações práticas de transmissão de conhecimentos, com ênfase no processo de análise de requisitos,

---

surge um questionamento que motiva a construção deste trabalho: dentro dos processos de Análise e Desenvolvimento de Sistemas, há a existência de requisitos que possam ser considerados essenciais e certamente aplicáveis para projetos de *software* em Tecnologia Assistiva?

Este trabalho apresenta um estudo sobre análise de requisitos de sistemas, com foco no estudo daqueles requisitos que possam compor uma base comum de verificações para as ações de analistas de sistemas em processos de desenvolvimento de *softwares* considerados como recursos de Tecnologia Assistiva para processos de análise de requisitos.

Além da motivação descrita, justifica-se também o desenvolvimento deste trabalho como oportunidade de construção de conhecimento que possa servir de base para mais pesquisas que busquem soluções de Tecnologia Assistiva. Ademais, proporciona conhecimentos complementares, acadêmicos e profissionais na formação em Sistemas de Informação. O trabalho está dividido entre os capítulos de introdução, revisão bibliográfica, desenvolvimento e considerações finais.

## **1.1 Objetivo Geral**

É objetivo geral deste trabalho obter uma visão sobre um possível conjunto de requisitos específicos e essenciais na análise de *softwares* que sejam caracterizados como recursos de Tecnologia Assistiva, mais especificamente para processos de análises de requisitos.

Foram verificados requisitos aplicáveis a *softwares* desenvolvidos para plataformas móveis (aplicativos e *smartphones*), com apreciação deste estudo por parte de uma pessoa com deficiência visual que já possui conhecimentos prévios sobre programação de computadores, chegando-se assim à possibilidade de gerar uma base de conhecimento sobre estes potenciais requisitos e sua aplicabilidade como padrões de *softwares* para deficientes visuais, dentro do conceito de Tecnologia Assistiva.

## **1.2 Objetivos específicos**

Para se contemplar o objetivo geral deste trabalho, foram realizadas as seguintes ações, correspondentes aos objetivos específicos a serem alcançados:

- Apresentação de conceitos sobre análise, técnicas e métodos de levantamento de requisitos de um sistema, por meio de pesquisa bibliográfica.
- Identificação e apontamento dos atributos e características de Tecnologias Assistivas em *softwares*.

- 
- Elaboração de uma proposta preliminar de requisitos que possam ser considerados essenciais para Tecnologias Assistivas, com ênfase em aplicativos para dispositivos móveis.
  - Aproximação prática do processo de análise dos requisitos com uso de metodologia ágil *SCRUM*, incluindo a participação externa de um *Product Owner* que detém propriedade de conhecimento ao contexto.
  - Propor um modelo para aplicação de possíveis requisitos essenciais ao desenvolvimento de um *software*, adequando essa adição ao momento da elicitação de requisitos no ciclo de vida da metodologia *SCRUM*.

### 1.3 Metodologia

Este trabalho foi construído partindo de uma abordagem qualitativa. Utilizou-se pesquisa exploratória de conteúdo relacionado ao tema principal proposto, além de experimentação de conceitos presentes em disciplinas na graduação em Sistemas de Informação.

A construção do conjunto de requisitos essenciais em Tecnologia Assistiva será feita com base nas premissas de análise de sistemas. Isso incluirá a consideração de um ambiente com aplicação de Metodologias Ágeis, para a elicitação daqueles requisitos considerados mais aderentes ao contexto. Além disso, contará com a apreciação das observações de uma pessoa com deficiência visual sobre estes requisitos, que participará como o *Product Owner*, da metodologia *SCRUM*.

Foi definido que o escopo de estudo dos requisitos correspondesse a plataformas de dispositivos móveis, caracterizados por *smartphones* e *tablets*, visto a abrangência desses dispositivos e sua representatividade relevante no acesso à Tecnologia da Informação. Também foi definido que as descobertas fizessem referência a desenvolvimentos voltados para deficientes visuais.

Os resultados do estudo são apresentados na forma de quadro, com características de fichas descritivas, com a síntese dos requisitos selecionados, formando assim a base de conhecimento pretendida por esse trabalho. Além de uma proposta para adequação do ciclo de vida tradicional do *SCRUM*, a fim de adaptar a inclusão dos requisitos essenciais não apenas em um único desenvolvimento, mas sim na própria metodologia.

## 2 Revisão Bibliográfica

### 2.1 Conceitos gerais sobre análise de sistemas

---

A primeira vez que se teve notícia sobre a origem de análise de sistemas, foi em 1940, quando uma corporação americana denominada *RAND Corporation*, que não tinha fins lucrativos, queria encontrar meios para ajudar nos problemas de planejamento de operações militares, mantendo o foco de suas pesquisas em análise de custos e em planejamento operacional, tático e estratégico (AUDY; ANDRADE; CIDRAL, 2009).

Pode-se dizer que a análise de sistemas se baseia em dois significados: o primeiro se trata de seu sentido mais amplo, que tem como sinônimos a pesquisa operacional, a análise de custo/benefício e a análise operacional. Consiste em uma aplicação do pensamento sistêmico na tentativa de solucionar problemas. Já o segundo significado, fundamenta-se em seu sentido mais restrito, que está associado ao desenvolvimento de sistemas de informação com uso de computadores e sistemas específicos, podendo ser utilizado para a construção de *softwares* comerciais, embarcados e o desenvolvimento de aplicativos para dispositivos móveis (AUDY; ANDRADE; CIDRAL, 2009).

De acordo com Audy, Andrade e Cidral (2009), a *RAND Corporation* possuía uma metodologia para gerenciar seus projetos de pesquisa, intitulada como Metodologia de Análise de Sistemas que observava os seguintes passos:

- I. Identificar os objetivos a serem alcançados.
- II. Buscar técnicas alternativas (ou sistemas) pelos quais os objetivos podem ser alcançados.
- III. Identificar os custos e os recursos necessários a cada alternativa (ou sistemas).
- IV. Desenvolver um ou vários modelos matemáticos, mostrando a interdependência dos objetivos, recursos, ambiente e as técnicas e instrumentos.
- V. Definir critérios relacionando objetivos e custos ou recursos, para a escolha de uma alternativa preferencial ou ótima.

Há, então, a possibilidade e compreensão de que a análise de sistemas seja uma análise de problemas e um estudo que está ligado aos sistemas de informação. Abrangendo a identificação dos objetivos da organização e a determinação do que deve ser feito para encontrar as soluções destes problemas, de modo que atenda melhor às necessidades entre os sistemas com o auxílio do levantamento de requisitos (BATISTA, 2017).

Levando-se em conta do que foi observado sobre os conceitos gerais sobre a análise de sistemas, requer-se uma atenção especial para que seja desenvolvido um trabalho compatível com a proposta de análise com foco em um *software* de Tecnologia Assistiva.

## **2.2 Técnicas para levantamento de requisitos de sistemas**

---

Primordialmente a Engenharia de Requisitos foi instituída como disciplina independente em 1993, durante o IEEE (*Institute of Electrical and Electronic Engineers*) e refere-se ao estudo do processo de produção e gerência dos requisitos que o *software* deverá cumprir, como o processo de descobrir, analisar, documentar e verificar serviços e restrições (OLIVEIRA et al., 2012).

Para Mazzola (2006), a plena compreensão de requisitos de *software* é determinante para o sucesso de um projeto de desenvolvimento de *software*. Caso a análise de requisitos seja mal efetuada isso definitivamente trará problemas para o cliente/usuário, mesmo que isso independa da exatidão que o *software* seja projetado e implementado.

Esses requisitos são imprescindíveis para os clientes, pois descrevem suas necessidades para o sistema e para o que ele irá servir, tal como, controlar um dispositivo, colocar um pedido ou encontrar informações. Os requisitos de *softwares* constituem em serem constantemente classificados como requisitos funcionais e não funcionais. (SOMMERVILLE, 2011).

De acordo com Machado (2018), existem técnicas para levantamentos de requisitos, que têm a finalidade de auxiliar o analista de sistemas e negócios a solucionar os problemas pertinentes. Estas técnicas possuem significados próprios, assim como apresentam vantagens e até mesmo desvantagens, que poderão ser utilizadas individualmente ou em conjunto pelo analista.

No contexto da análise de requisitos, é fundamental explorar diversas estratégias para coletar dados significativos. Uma abordagem comumente utilizada é a amostragem, como descrito por Carvalho (2009), que envolve um processo de investigação para obter dados em uma pequena parte de uma população. Os dados coletados a partir de uma amostra são, então, transformados em informações representativas sobre a determinada população.

A investigação é outro processo de levantamento de requisitos que envolve a obtenção e avaliação de documentos para buscar compreensão sobre a regra em análise. Há também a estratégia da entrevista, realizada através de conversas direcionadas a entrevistados que podem expressar opiniões sobre a organização, a regra de negócio e processos aos quais um *software* precisa corresponder (CARVALHO, 2009).

A aplicação de questionários permite que o levantamento de requisitos seja feito atingindo-se um grande número das pessoas que serão afetadas pelo sistema em desenvolvimento. Um questionário pode quantificar as visões que os envolvidos no processo de levantamento de requisitos possuem para o *software* em desenvolvimento (CARVALHO, 2009).

---

Outra técnica que demonstra eficácia na identificação dos requisitos é a estratégia de observação, onde o comportamento dos usuários e o ambiente onde o *software* residirá é objeto de estudo presencial por parte do analista, que documenta suas percepções sobre o processo observado para usar esta informação como fonte de conhecimento dos requisitos. Finalmente, há a prototipação do próprio *software* em desenvolvimento, que busca oferecer algum nível mais concreto de abstração sobre o produto final que se encontra em análise (CARVALHO, 2009).

Independente da estratégia adotada, infere-se que qualquer desenvolvimento de *software*, incluído aí o contexto da Tecnologia Assistiva, depende muito da compreensão acerca das necessidades que o próprio *software* busca suprir. Desta forma, a importância do levantamento de requisitos sempre terá como finalidade sistematizar o processo de definição dessas técnicas e proporcionar uma base de conhecimento sobre o sistema que deve garantir sua qualidade.

### 2.2.1 Requisitos funcionais

Os requisitos funcionais são os que caracterizam o que um sistema deve fazer, dependendo do tipo de *software* a ser desenvolvido, de quem possa ser seus possíveis usuários e da conduta adotada pela organização que tem como finalidade escrevê-los. Ao se tratar sobre requisitos de usuário, os requisitos funcionais são geralmente descritos de forma genérica para que facilite a compreensão dos usuários do sistema. Por outro lado, os requisitos funcionais de sistemas mais específicos, relatam funções dos sistemas, como suas entradas e saídas (SOMMERVILLE, 2011). O autor conclui:

Em princípio, a especificação dos requisitos funcionais de um sistema deve ser completa e consistente. Completude significa que todos os serviços requeridos pelo usuário devem ser definidos. Consistência significa que os requisitos não devem ter definições contraditórias. Na prática, para sistemas grandes e complexos, é praticamente impossível alcançar completude e consistência dos requisitos. Uma razão para isso é que ao elaborar especificações para sistemas complexos é fácil cometer erros e omissões. (SOMMERVILLE, 2011, p. 59).

### 2.2.2 Requisitos não-funcionais

Os requisitos não-funcionais, de acordo com Sommerville (2011, p. 60), se baseia no seguinte conceito:

São requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área. Uma alternativa a esse cenário seria os

---

requisitos definirem restrições sobre a implementação do sistema, como as capacidades dos dispositivos de E/S ou as representações de dados usadas nas interfaces com outros sistemas. (SOMMERVILLE, 2011, p.60)

O desempenho, a proteção ou a disponibilidade, conhecidos como requisitos não-funcionais, geralmente especificam ou limitam os atributos do sistema como um todo, sendo assim, requisitos mais críticos. Os usuários, podem, de certa maneira, encontrar jeitos de burlar uma função do sistema que realmente não atenda a suas necessidades, mas no momento em que isso ocorre - não atender à um requisito não-funcional, faz com que aconteça a inutilização de todo o sistema (SOMMERVILLE, 2011).

Ademais Sommerville (2011) afirma que a dificuldade de separar os requisitos funcionais e não funcionais faz com que o relacionamento entre eles complique a forma de serem compreendidos. Contudo, os requisitos que se relacionam com as propriedades decorrentes do sistema, como o desempenho ou a confiabilidade, devem ficar evidentes.

### **2.3 Desenvolvimento de sistemas para dispositivos móveis**

O avanço da computação móvel e ubíqua resultou da redução do tamanho dos dispositivos e da melhoria na conectividade sem fio. À medida que os aparelhos diminuíram de tamanho, ganharam em autonomia e deixaram de depender de tomadas, enquanto adquiriram novos recursos, tornou-se mais prático carregá-los consigo ou incorporá-los ao vestuário. Simultaneamente, à medida que as conexões sem fio se tornaram mais comuns, mais rápidas e confiáveis, tornou-se viável interligar esses dispositivos entre si. Por exemplo, um smartwatch pode tirar proveito da capacidade expandida de um smartphone, permitindo uma interação mais fluida com ambos (SIMAS; BORGES; COUTO; et al, 2019).

De acordo com Junior (2018), a alta demanda torna o desenvolvimento de aplicações (APPs) para esses dispositivos altamente objetivado. Desta maneira, consegue-se acompanhar seu constante desenvolvimento, novas tecnologias e procedimentos que aparecem frequentemente.

Junior (2018) diz que os produtos de *software* precisam ter qualidade. Ele afirma que “[...] é necessário ter um processo de desenvolvimento com qualidade, bem como com custos controlados e prazos bem estipulados”. De acordo com Fuggeta (2000) citado por Junior (2018), esta qualidade do *software* requer “um conjunto de políticas, estrutura organizacional, tecnologias, procedimentos e artefatos necessários para conceber, desenvolver, implantar e manter produtos de *software*”.

---

Os aplicativos móveis, frequentemente referidos como APPs, são programas concebidos e criados com a finalidade de operarem de maneira especializada em aparelhos eletrônicos móveis, como *Palmtops*, *Tablets*, dispositivos de reprodução de música em formato mp3, telefones celulares e *smartphones* mais recentes, que dispõem de significativa capacidade de armazenamento e poder de processamento (SIMAS; BORGES; COUTO; et al, 2019).

Os aplicativos são disponibilizados para *download* através de lojas virtuais como a *Apple Store* – loja virtual da *Apple*, *Play Store* – loja virtual do *Google* para dispositivos móveis com o sistema operacional *Android*, *Windows Phone Store* – loja virtual da *Microsoft* para *Windows Phone*, entre outras diversas lojas compatíveis com seus respectivos sistemas operacionais, sendo possível encontrar todos os tipos e gêneros de aplicações (SILVA; PIRES; NETO, 2015).

Silva, Pires e Neto (2015) relatam que o mercado de dispositivos móveis é envolto por diversos fabricantes, o que favorece na inclusão de múltiplas plataformas para desenvolvimento, sistemas operacionais móveis, *software* e *hardware*. Essa grande quantidade de plataformas faz com que tenha uma variedade enorme de aplicativos, sendo cada um codificado para ser executado através de uma arquitetura específica. Entretanto, isso se torna um dos principais desafios da computação móvel, pois, para cada sistema operacional, deve haver uma nova aplicação (MARTINS et al., 2013 apud SILVA; PIRES; NETO, 2015).

No processo de desenvolvimento de um aplicativo móvel, a escolha do tipo de aplicativo é um ponto muito importante (MARTINS et al., 2013 apud SILVA; PIRES; NETO, 2015). Para se iniciar um projeto necessita-se da realização de uma análise detalhada e estratégica sobre a plataforma, sistemas, produtos e arquiteturas que possam ser utilizadas (SILVA; PIRES; NETO, 2015).

De acordo com Silva, Pires e Neto (2015, p. 26), no início de um projeto, estima-se que os principais aplicativos móveis são:

Aplicativos *Web Apps* (sites móveis), aplicativos móveis nativos e aplicativos híbridos, que consistem respectivamente em soluções feitas para *Web* formatadas para serem acessadas através do *browser* dos dispositivos móveis, soluções que são desenvolvidas para um delimitado dispositivo móvel e sistema operacional e aplicações desenvolvidas com a junção do nativo e *Web* (SILVA; PIRES; NETO, 2015, p.26).

Observa-se que a importância e a relevância dos aplicativos móveis no contexto do *software* na atualidade, e considerando a crescente busca pela qualidade destes aplicativos, pode-se então, ser inserida nessa busca a preocupação para com o desenvolvimento de aplicativos que atendam às necessidades de tecnologias assistivas.

---

## 2.4 Características de Tecnologias Assistivas

Um estudo feito pela Organização das Nações Unidas, estima que cerca de 10% da população mundial – aproximadamente 650 milhões de pessoas – possuem algum tipo de deficiência (WHO, 2012 apud MIZRAHI; CÍCERO 2016). No Brasil, o Censo de 2010 mostra que 23,9% da população – cerca de 45 milhões de pessoas – dos 190 milhões de brasileiros, vive com alguma deficiência (IBGE, 2012 apud MIZRAHI; CÍCERO 2016).

A inclusão dessas pessoas com deficiências, enquanto processo social tem sido bastante evidenciado no Brasil e no mundo nas últimas duas décadas (PASSERINO; MONTARDO, 2007). Conforme Passerino e Montardo (2007) a inclusão é um processo que se estabelece dentro de uma sociedade mais ampla que tem o intuito de atender as necessidades relacionadas a qualidade de vida do ser humano, o seu desenvolvimento, a autonomia de sua renda e a equidade de oportunidades e direitos para os indivíduos e grupos sociais que em algum momento da sua vida se veem em situações de desvantagem com relação aos outros membros da sociedade.

Do mesmo modo que a inclusão social, Reinaldi, Júnior e Calazans (2011, p.2) referência que a inclusão digital abrange a introdução de todas as pessoas, deficientes ou não, no mundo virtual. E complementa que “o processo de inclusão do indivíduo portador de deficiência é complexo, o processo de sua inclusão no mundo digital apresenta uma série de dificuldades e obstáculos”. Através da Tecnologia da Informação é possível promover a inclusão digital por meio da Tecnologia Assistiva.

A expressão Tecnologia Assistiva surge pela primeira vez na década de 80 e sua definição ainda se encontra em construção, sendo assim, é possível encontrar diferentes conceitos em relação a esse termo (OLIVEIRA; MILL, 2016). Mas deve ser entendida como um auxílio que proporcionará o aumento de uma habilidade funcional deficitária ou facilitará a realização da função que o indivíduo deseja, mas se depara impedido por motivo de uma deficiência ou pelo envelhecimento (BERSCH, 2017).

Para Bersch (2017, p.2), o maior objetivo da tecnologia assistiva é proporcionar à pessoa com deficiência “maior independência, qualidade de vida e inclusão social, através da ampliação de sua comunicação, mobilidade, controle de seu ambiente, habilidades de seu aprendizado e trabalho”.

Dadas as nuances citadas dentro do objetivo da tecnologia assistiva, seus recursos classificados ou organizados de acordo com os objetivos funcionais a quais foram destinados. Essas classificações foram criadas com finalidades distintas e ao apresenta-las, destaca-se a sua

---

importância na organização da utilização, prescrição, estudo e pesquisa de recursos e serviços em TA, assim como a formação de recursos humanos nesta área (BERSCH, 2017).

Bersch (2017) afirma ainda que para a boa compreensão da tecnologia assistiva, é importante considerá-la como um recurso do usuário e não apenas como recurso profissional, isto é, justifica pelo fato de que ela serve à pessoa com deficiência que necessita desempenhar funções do cotidiano de forma independente.

Nota-se que a inclusão de pessoas deficientes através dessas tecnologias é essencial, sobretudo para os deficientes visuais. Uma vez que, os dispositivos móveis tem uma participação significativa como uma ferramenta de comunicação entre eles, possibilitando uma maior autonomia e contribuindo para que tenham uma qualidade de vida com maior satisfação e participação social (BERSCH, 2013 apud SANTOS et al 2017).

Levando em consideração esses aspectos, a inclusão digital por meio da tecnologia assistiva e sua importância para pessoas com deficiência, permite que a sua compreensão e utilização em diferentes ambientes e situações se dá por meio de diversos equipamentos e navegadores, como por exemplo, *softwares* para dispositivos móveis.

## **2.5 Necessidades especiais para deficientes visuais**

O conceito de pessoa com deficiência adotado constitucionalmente no Brasil transcende o aspecto estritamente clínico e assistencialista, realçando o fator político para o reconhecimento da necessidade de superar barreiras sociais, políticas, tecnológicas e culturais. Pessoas com limitações físicas, mentais, intelectuais ou sensoriais compartilham atributos equiparáveis a qualquer outra diversidade humana, destacando que a deficiência está mais relacionada às barreiras sociais do que a atributos inerentes, enfatizando a necessidade de recursos que garantam a extensão completa dos direitos humanos para todas as pessoas com deficiência (FERRAZ; LEITE; LEITE, 2012).

O Decreto nº 3.298/99 juntamente com o Decreto nº 5.296/04 que foi responsável pela a inclusão de pessoas com deficiência de baixa visão na seguinte lista que caracteriza a deficiência visual como:

- Cegueira – com a melhor correção óptica, a visual é igual ou menor que 0,05 no melhor olho.
- Baixa Visão - a percepção visual no melhor olho é de 0,3 e 0,05, com a melhor correção óptica.
- Quando a somatória da medida do campo visual em ambos os olhos for igual ou inferior que 60°.

- 
- Quando as condições anteriores ocorrem de forma síncrona.

Reinaldi, Júnior e Calazans (2011, p. 39-40), ainda relatam que as pessoas são identificadas como deficientes visuais pelos seguintes motivos:

Mesmo usando óculos comuns, lentes de contato ou implantes de lentes intraoculares, não conseguem ter uma visão nítida, podendo ter sensibilidade ao contraste, percepção das cores e intolerância à luminosidade, conforme o tipo de patologia causadora da perda visual (REINALDI; JÚNIOR; CALAZANS, 2011, p.39-40).

Para a inclusão das pessoas com deficiência visual no meio da tecnologia, elas têm como garantia a interação adequada com os dispositivos móveis, através de uma série de ferramentas de acessibilidade disponíveis por meio dos sistemas operacionais, como por exemplo, leitor de tela, lente de aumento e controle de contraste (LEPORINI et al., 2012 apud SILVA; DAMACENO; BRAGA, 2015).

Rinaldi, Júnior e Calazans (2011) também informam que a facilidade de perceber como construir uma *interface* para deficientes visuais é algo complexo:

Deve seguir algumas regras de lógica, funcionalidade e ergonomia muito particulares, e que serão parte fundamental no sucesso de um produto. É importante notar que não é suficiente uma adaptação, uma *interface* já padronizada, bastante conhecida e consolidada, pois isso não garante que ela seja acessível às pessoas cegas (REINALDI; JÚNIOR; CALAZANS, 2011, p. 43).

De acordo com o Grupo GUIA (1999), citado por Conforto e Santarosa (2002) existe alguns problemas enfrentados pelos usuários cegos em relação a acessibilidade *web*, como: adquirir informações apresentadas visualmente, usar dispositivo diferente do teclado para interagir, navegar por meio de conceitos espaciais e especificar outros sons da voz produzida pelo sintetizador.

Outros aspectos que se referem às características citadas acima, é quando o sistema deve ser acessível e utilizável em diversas situações, independente do *software*, da comunicação ou do equipamento, como a compatibilidade com navegadores; a utilização de comunicações lentas ou de equipamentos sem saída de áudio; o uso de equipamentos sem mouse e o manuseio de impressoras monocromáticas (GRUPO GUIA, 1999 apud CONFORTO; SANTAROSA, 2002).

O Grupo GUIA (1999), citado por Conforto e Santarosa (2002) acrescenta que as características de ambiente também afetam muito na utilização de sistemas para a acessibilidade *web*, isto é, o acesso não deve ser realizado pelo ambiente físico envolvente, exterior ou interior, com ruídos ou com muita luminosidade.

---

Assim, entende-se que a acessibilidade e a inclusão digital de deficientes visuais não se referem somente ao acesso a redes de informações, mas também à eliminação de obstáculos como a comunicação, equipamentos e *softwares* adequados para atender as diferentes necessidades desses indivíduos. Pois, as tecnologias assistivas juntamente com o desenvolvimento tecnológico colaboram para que essas pessoas obtenham a inclusão, independência e progresso.

## 2.6 Metodologias Ágeis

Quando uma empresa procura alternativas para às abordagens tradicionais para o desenvolvimento de *software*, as metodologias ágeis aparecem como uma solução (SOARES, 2004). O termo surgiu em fevereiro de 2001, em um resort de ski, em Utah, nos Estados Unidos, quando um grupo com dezessete especialistas se reuniram para articular sobre as boas práticas adotadas para o desenvolvimento de seus trabalhos de uma forma mais leve, rápida e centrada (JUNIOR, 2020).

De acordo com Junior (2020), o grupo que representava os métodos *Extreme Programming* (XP), *SCRUM*, *DSDM*, *Crystal* e outros, chegaram a um consenso sobre os pontos mais importantes para o desenvolvimento de *software* e estabeleceram valores e princípios comuns, que foi intitulado como manifesto do desenvolvimento de *software* ágil. Sobre os novos valores do desenvolvimento de *software*, Junior (2020) relata que:

Estamos descobrindo maneiras melhores de desenvolver *software* fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através desse trabalho, passamos a valorizar: 1 - Indivíduos e interação mais que processos e ferramentas; 2 - *Software* funcionando mais que documentação abrangente; 3 - Colaboração com o cliente mais que negociação de contratos; 4 - Responder a mudanças mais que seguir um plano. Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda (JUNIOR, 2004, p. 27-28).

Conforme Soares (2004), o Manifesto Ágil, mostra que os processos e ferramentas, a documentação, a negociação de contratos ou planejamento, têm importância secundária quando se compara com os indivíduos e interações, com o *software* estar a funcionar, com o auxílio do cliente e as respostas rápidas a mudanças e alterações. Essas definições aproximam-se melhor com a maneira que pequenas e médias organizações trabalham e respondem a mudanças. Entre as metodologias ágeis, a que abordaremos nesse trabalho é a Metodologia *SCRUM*.

Em meados de 1986, quando o ritmo do desenvolvimento de novos produtos estava cada vez mais acelerado e competitivo, a velocidade e a flexibilidade eram essenciais. Deste modo, dois professores universitários japoneses, Hirotaka Takeuchi e Ikujiro Nonaka

---

publicaram um artigo na *Harvard Business Review* denominado ‘*The New Product Development Game*’ - O novo jogo para o desenvolvimento de produtos (JUNIOR, 2020).

Este novo método de desenvolvimento concedia que as empresas pudessem desenvolver seus produtos com maior rapidez e flexibilidade. Seus processos eram aplicados para equipes multifuncionais que possuíam autonomia e autoridade para tomada de decisões. De 1993 em diante, Jeff Sutherland e Ken Schwaber passaram a se concentrar no desenvolvimento *SCRUM*, publicando em 2010, a primeira versão do Guia *SCRUM*, onde contém o conceito e as regras do Framework *SCRUM* (JUNIOR, 2020).

Junior (2020), diz que o método *SCRUM* é utilizado para desenvolver, entregar e manter produtos complexos, ele define que “é um *framework* dentro do qual as pessoas podem tratar e resolver problemas complexos e adaptativos. É importante reforçar que o *SCRUM* é um *Framework* estrutural, e não uma técnica, um processo ou uma metodologia” (JUNIOR, 2020, p. 46-47).

De acordo com Junior (2020), o *SCRUM* divide o desenvolvimento em *Time Sprint* (interações), onde ocorre o empenho para o desenvolvimento do produto, que tem no máximo trinta dias de duração. Em seus artefatos (*Backlog* do Produto, *Backlog* do *Sprint* e Incremento), que retratam o trabalho a ser executado, fornecendo transparência e oportunidades para realizar a inspeção e adaptação, quando se está sendo produzido ou entregue.

Para Machado e Medina (2009), outro fator importante para o desenvolvimento *SCRUM* é o *Time SCRUM* que tem como responsabilidade a entrega de produtos de forma iterativa e incremental. *SCRUM Team* é uma equipe de desenvolvimento composta geralmente por cinco a nove membros e tem como função transformar os itens do *Product Backlog* em itens do *Sprint Backlog* e convertê-los em *software* pronto para ser entregue.

O *SCRUM Master* tem a missão de fazer o *SCRUM Team* funcionar e aplicar corretamente o método *SCRUM*, garantindo que seja um time totalmente funcional e produtivo. Como executa o papel de gerenciamento, ele protege a equipe de interferências externas e garante que os *Sprints* não possuam itens que não possam ser entregues (MACHADO; MEDINA, 2009).

Ainda de acordo com Machado e Medina (2009), o *Product Owner* é conhecido em algumas empresas como Gerente de Produtos, sendo o representante de todos os clientes, um especialista de negócios. Como é ele quem sabe o que é mais importante para o desenvolvimento, tem como função criar e priorizar o *Release Plan* e o *Product Backlog*, mantendo a sua atualização. Portanto, o papel do *Product Owner* será imprescindível para a descoberta dos requisitos que são o objetivo deste trabalho.

---

### 3 DESENVOLVIMENTO

O levantamento dos requisitos de um sistema é parte fundamental para o bom resultado na construção de um *software*, além de Ilustração como etapa inicial em processos de ciclo de vida de metodologias de desenvolvimento, como é o caso da Metodologia Ágil *SCRUM*.

Os resultados obtidos através de um levantamento de requisitos adequado são utilizados como guias para a equipe de desenvolvimento, havendo inclusive a oportunidade de se gerar padrões de desenvolvimentos baseados em requisitos que podem se apresentar de forma recorrente em análises de *softwares* dentro de um mesmo domínio (PRESSMAN, 2016).

#### 3.1 Perfil do Product Owner

O *Product Owner*, convidado para participar deste estudo, exerce profissionalmente a função de gestor em telefonia, com grande experiência no mercado de tecnologia da informação. Este profissional também é uma pessoa com deficiência visual e depende do uso da tecnologia para cumprir suas tarefas do trabalho, além de enfrentar situações onde a acessibilidade tende a não favorecer um desenvolvimento adequado de suas atividades profissionais.

O convidado possui conhecimento técnico que foi adquirido através de estudos próprios, e tem uma percepção privilegiada sobre as necessidades que devem ser atribuídas a *softwares* que busquem atuar no âmbito da Tecnologia Assistiva, uma vez que seu cotidiano está relacionado a este contexto. Justamente por isso, o *Product Owner* detém o potencial de tecnologia para o desenvolvimento deste trabalho.

Além disso, o *Product Owner* reconhece que a Tecnologia da Informação de uma forma geral ajuda com o trabalho que ele exerce e que a tecnologia está inserida neste contexto para favorecer o cenário deste profissional que como *Product Owner* representa um segmento da sociedade que depende de acessibilidade em *softwares*.

Como contribuição a este trabalho, o *Product Owner* narrou necessidades que são perceptivelmente indispensáveis na fase de desenvolvimento de um *software* que se propõe a apresentar características de Tecnologia Assistiva. Como o caso específico do *Product Owner* é a deficiência visual, serão consideradas estas características, não havendo aplicação direta e imediata para outro tipo de deficiência (auditiva), porém a premissa do trabalho poderá ser mantida também nessas situações.

#### 3.2 Adições assistivas na análise de requisitos

---

Considerando o levantamento tradicional de requisitos de um sistema, havendo também a organização destes requisitos em um modelo de trabalho baseado na Metodologia Ágil *SCRUM*, tem-se a criação de um *Product Backlog*, com a relação de todas as necessidades elencadas pelo *Product Owner*. Essas necessidades serão incluídas em um *Sprint* de acordo com as decisões do *SCRUM Team*.

Na proposta de Tecnologias Assistivas, cada uma dessas necessidades passará por adições que irão agregar valor de acessibilidade ao desenvolvimento. Através das contribuições do *Product Owner* serão consideradas as seguintes adições:

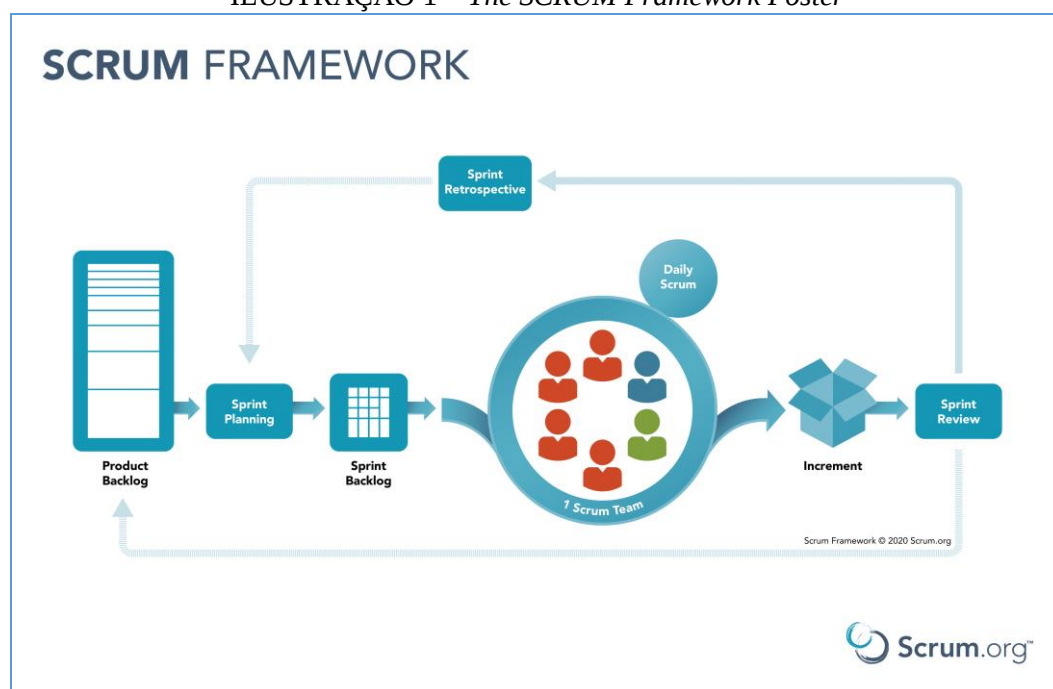
- Elementos de interface devem ser lidos para prover informações ao usuário, portador de deficiência visual, sobre a própria interface e suas ações;
- Ações existentes no *software* podem oferecer alertas ao usuário, funcionando como lembretes de compromissos ou de execuções cotidianas do *software*;
- Personalizações parametrizáveis;
- Explorar sensações associadas a outros sentidos;
- *Feedback*, sempre informar o que vai acontecer;
- Confirmação, sempre confirmar em casos críticos;
- Sempre perguntar se quer perguntar isso novamente;
- Previsibilidade de ações, aprender com o usuário (janelas e mais janelas);
- Alto acoplamento.

Além das adições apresentadas, deve-se partir de um pressuposto que o controle por voz é obrigatório, assim como, a leitura de tela, se tornado então, a sua existência algo inevitável para o programa.

### **3.3 Artefatos para os requisitos**

Como apresentando anteriormente no capítulo dois, subseção 2.6 – Metodologias Ágeis, foram relatados os artefatos da metodologia *SCRUM*. A Ilustração 1 demonstra uma visão gráfica de como esses artefatos são implementados a nível de equipe dentro de uma organização.

ILUSTRAÇÃO 1 – *The SCRUM Framework Poster*



Fonte: SCRUM.org.

De acordo com a Ilustração 1, a metodologia *SCRUM* preza por uma organização para executar os seus métodos, começando primeiramente com a lista ordenada (*Product Backlog*) de tudo o que é conhecido ser necessário para a criação do produto. Em seguida, a equipe seleciona esses itens listados que contém um plano (*Sprint Planning*) a ser utilizado para a entrega do produto e atingir os objetivos do *Sprint Backlog*.

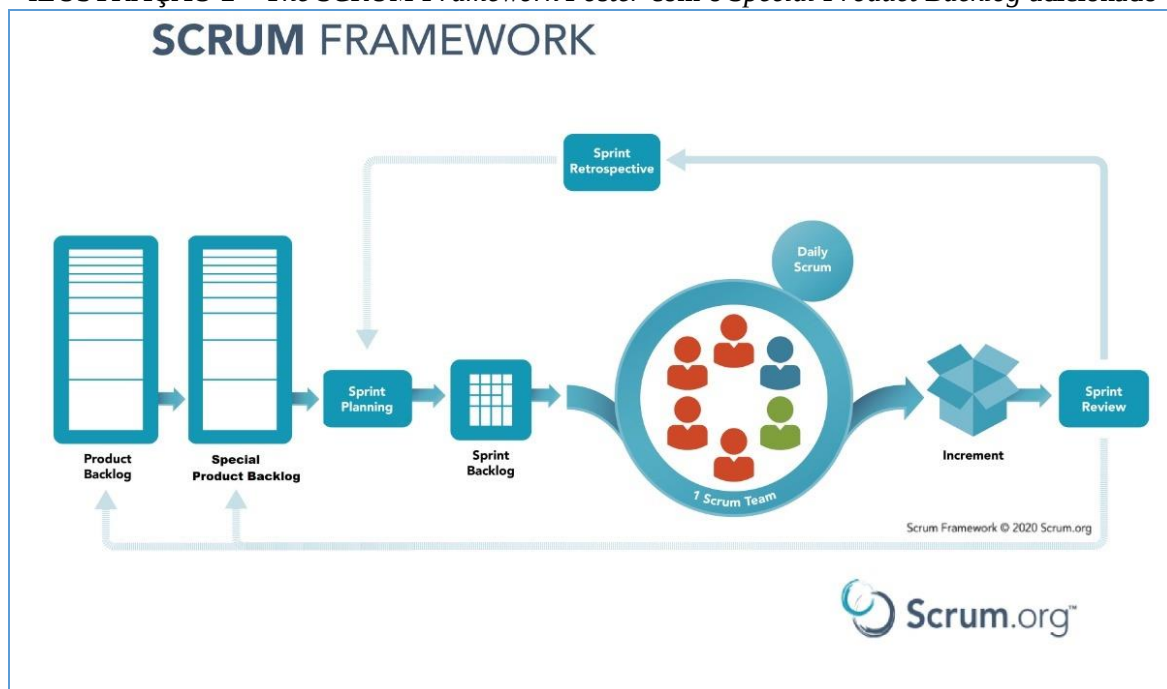
Dentre essas funções, sempre irá ocorrer reuniões com o *SCRUM Team* durante todos os dias, além das reuniões com o *Product Owner* de duas a quatro semanas. As reuniões tem como intuito revelar se o produto precisa de alteração ou não para poder seguir em frente com a sua execução. Dito isto, com a concordância da equipe e do *Product Owner* todos os itens do *Product Backlog* que foram selecionados e finalizados na *Sprint* se torna um incremento (*Increment*).

Este incremento poderá passar por uma revisão (*Sprint Review*) e caso não esteja de acordo com o que a equipe e o *Product Owner* requerem, pode passar por uma retrospectiva (*Sprint Retrospective*) e assim ser analisado novamente. Ou pode ser considerado completo, sendo denominado com a Definição de Pronto (JUNIOR, 2020).

Como o trabalho propõe a análise de requisitos assistivos, deve-se então incluir na execução desses métodos o *Special Product Backlog*, que tem como objetivo separar os itens que são assistivos para a criação de um produto inclusivo. Em síntese, todos os requisitos devem-se somar as adições assistivas, como forma de um *checklist* que resultará no *Special*

*Product Backlog* que deve ficar entre o *Product Backlog* e o *Sprint*. Isto é, só irá para o *Sprint* (desenvolvimento) os requisitos que forem aprovados nas duas fases: o requisito original e o requisito assistivo (Ilustração 2).

ILUSTRAÇÃO 2 – The SCRUM Framework Poster com o *Special Product Backlog* adicionado



Fonte: SCRUM.org (adaptado).

Propõe-se, portanto, a utilização de um *checklist* de adições assistivas aos requisitos, como novo artefato para análise de requisitos essenciais em Tecnologia Assistiva. Este artefato é apresentado pelo Quadro 1, onde o requisito original, que será descrito no *Product Backlog*, contará com um título, um breve detalhamento, uma identificação (número) de quantos requisitos o produto contém, e deverá marcar o seu tipo: funcional ou não-funcional. Deve-se marcar também, quantos dias ou horas precisará para o desenvolvimento deste produto.

Caso o produto seja inclusivo, é necessário que faça um *checklist* dos requisitos assistivos (*Special Product Backlog*) mencionados no Quadro 1, que poderão ser personalizados. Marca-se também, o tempo necessário que irá levar para o seu desenvolvimento, que se somará ao tempo total do desenvolvimento do requisito original.

QUADRO 1: Artefato para *checklist* de adições assistivas aos requisitos

| Quadro 1: Alerta para checklist de adições assistivas aos requisitos                                          |                                     |    |  |                                       |          |
|---------------------------------------------------------------------------------------------------------------|-------------------------------------|----|--|---------------------------------------|----------|
| Requisito original                                                                                            |                                     | ID |  | Tipo                                  | o F o NF |
| Detalhamento                                                                                                  |                                     |    |  |                                       |          |
| Tempo necessário para desenvolvimento                                                                         |                                     |    |  | o D. o H.                             |          |
| ADIÇÃO DE REQUISITOS ASSISTIVOS                                                                               |                                     |    |  |                                       |          |
| Requisito                                                                                                     |                                     |    |  | Tempo necessário para desenvolvimento |          |
|                                                                                                               | Alerta com temporizador             |    |  | o D. o H.                             |          |
|                                                                                                               | Conteúdo de descrição funcional     |    |  | o D. o H.                             |          |
|                                                                                                               | Conteúdo de descrição não-funcional |    |  | o D. o H.                             |          |
|                                                                                                               | Feedback antecipado de confirmação  |    |  | o D. o H.                             |          |
|                                                                                                               | Manutenção para ações futuras       |    |  | o D. o H.                             |          |
| ACOPLAMENTOS                                                                                                  |                                     |    |  |                                       |          |
| Próximas ações decorrentes deste requisito                                                                    |                                     |    |  |                                       |          |
|                                                                                                               |                                     |    |  |                                       |          |
| Legenda:<br>ID – Identificação F – Requisitos Funcionais NF – Requisitos Não-Funcionais<br>D – Dias H – Horas |                                     |    |  |                                       |          |

Fonte: autoria própria.

Os acoplamentos serão descritos pelo próprio desenvolvedor, como por exemplo, a chamada de uma nova classe ou as próximas ações do banco de dados a serem executadas, a partir do requisito original. Iterativamente, estes acoplamentos serão incrementados de acordo com a validação dos requisitos, permitindo que o desenvolvimento mapeie ações adjacentes, com o objetivo de antecipá-las, para favorecer sua usabilidade.

### 3.4 *Special Product Backlogs*

Existem questões de autoestima envolvidas com relação à boa apresentação de uma interface assistiva, de forma que não existam julgamentos errôneos quanto à Tecnologia Assistiva não ser agradável aos olhos de quem vê. Além de uma interface assistiva, entra em

---

questão também a explicação da função do botão existente no sistema, de uma maneira metafórica, envolvendo sentidos e não cores (FERRAZ; LEITE; LEITE, 2012).

A utilização de um sistema tecnológico mais humanizado, se torna importante, pois para uma pessoa com deficiência, os outros sentidos naturais ficam mais aguçados. Isso deve ser explorados pelos desenvolvedores, criando um sistema capacitado para atender as necessidades desses usuários.

Para que se tenha *softwares* mais inclusivos, é necessário o auxílio dos requisitos assistivos para o desenvolvimento do mesmo, portanto, é fundamental que utilize como parâmetro o *Special Product Backlog* para a construção desses sistemas. Que tem o intuito de garantir que o usuário com deficiência tenha uma boa experiência no ambiente tecnológico, com qualidade e dignidade para exercerem o seu direito de inclusão.

#### 4 CONSIDERAÇÕES FINAIS

Este trabalho abordou propostas para requisitos essenciais de tecnologia assistiva, e foram verificados conteúdos associados a análise de requisitos com observações que levam a sua compreensão voltada à tecnologia assistiva. Identificou-se, através dessas verificações, técnicas de levantamento de requisitos funcionais e não-funcionais que foram considerados relevantes na construção de sistemas assistivos.

Para que o trabalho não se limitasse à teoria, elaborou-se uma proposta preliminar de requisitos que auxiliam na construção de aplicativos para dispositivos móveis inclusivos através da metodologia ágil *SCRUM*, que tem como objetivo estabelecer princípios de trabalho no ciclo de vida de desenvolvimento de um *software*.

Destaca-se o papel do *Product Owner* neste trabalho, por ser uma pessoa com deficiência visual, Ilustração na contribuição de relatos que foram essenciais na fase de desenvolvimento, por obter conhecimento técnico sobre a tecnologia e ser um profissional liberal, ele necessita da tecnologia para a realização de suas tarefas cotidianas.

Desta forma, constatou-se através desses relatos, a necessidade de criar um *Product Backlog*, como forma de acrescentar requisitos assistivos no ciclo de vida do método *SCRUM*. Portanto, a última parte do trabalho foi empenhada em construir um quadro para a realização de um *checklist* para facilitar a criação de um sistema assistivo, denominando-se *Special Product Backlog*, que tem o objetivo de facilitar a inclusão digital e a acessibilidade de pessoas com deficiência visual, tornando-o mais humanizado para atender às suas necessidades.

A adição do *Special Product Backlog* provocou a necessidade de interferir no modelo do ciclo de vida do *SCRUM*, com a inclusão desta nova etapa imediatamente anterior à

---

definição do *sprint* de desenvolvimento. Isso representa uma mudança de paradigma na metodologia que pode ser explorada quanto à sua viabilidade em novos estudos.

Portanto, considera-se que há a possibilidade de existir requisitos com potencial em ser essenciais e, certamente, aplicáveis para projetos de *softwares* em Tecnologia Assistiva. Ressalta-se que este trabalho teve como foco a aproximação da análise para um sistema destinado ao uso de uma pessoa com deficiência visual. Entretanto, cabe afirmar que a premissa da adição do *Special Product Backlog*, bem como a dinâmica de sua construção, possa ser feita observando-se particularidades no desenvolvimento de *softwares* que irão atender a outros usuários portadores de deficiência diferente da visual.

Como proposta para trabalhos futuros sugere-se uma possível aplicação prática do artefato de análise apresentado neste trabalho a fim de avaliar sua aplicabilidade prática no contexto do ciclo de vida de desenvolvimento de um *software* de tecnologia assistiva, bem como sua adaptação a outros contextos de desenvolvimento que contemplem adições específicas para outros tipos de deficiência.

## REFERÊNCIAS

AUDY, Jorge Luis Nicolas; ANDRADE, Gilberto Keller; CIDRAL, Alexandre. **Fundamentos de sistemas de informação**. Bookman editora, 2009.

BATISTA, Emerson de Oliveira. **Sistemas de informação**. Saraiva Educação SA, 2017.

BERSCH, Rita. **Introdução à tecnologia assistiva**. Porto Alegre: CEDI, v. 21, 2017.

CAMPOS, Flavio R.; BLIKSTEIN, Paulo. **Inovações radicais na educação brasileira. (Tecnologia e Inovação na Educação Brasileira)**. Porto Alegre: Grupo A, 2019.

CARVALHO, Pedro F.. **Técnicas de Levantamento de Requisitos**. 2009. Disponível em: <[http://pedrofcarvalho.com.br/PDF/ENGENHARIA\\_ANALISE\\_LEVANTAMENTO\\_REQUSITOS\\_2.pdf](http://pedrofcarvalho.com.br/PDF/ENGENHARIA_ANALISE_LEVANTAMENTO_REQUSITOS_2.pdf)> Acesso em: 24 ago. 2020.

CONFORTO, Débora; SANTAROSA, Lucila. Acessibilidade à web: internet para todos. **Informática na educação: teoria & prática**, v. 5, n. 2, 2002.

DE OLIVEIRA, Camila Dias; MILL, Daniel. Acessibilidade, inclusão e tecnologia assistiva: um estudo bibliométrico. **Revista Ibero-Americana de Estudos em Educação**, v. 11, n. 3, p. 1169-1183, 2016.

D'OLIVEIRA, André Luiz Câmara et al. Breve ensaio sobre a evolução histórica da engenharia de requisitos: motivações para seu surgimento. **Universitas: Gestão e TI**, v. 2, n. 2, 2012.

DOS SANTOS SOARES, Michel. Comparação entre metodologias Ágeis e tradicionais para o desenvolvimento de *software*. **INFOCOMP Journal of Computer Science**, v. 3, n. 2, p. 8-13, 2004.

---

FERRAZ, Carolina V.; LEITE, George S.; LEITE, Glauber S.; et al. **Manual dos direitos da pessoa com deficiência**. São Paulo: Editora Saraiva, 2012.

JUNIOR, Heleno de Souza Campos. Desenvolvimento de aplicações para dispositivos móveis. **ANALECTA-Centro de Ensino Superior de Juiz de Fora**, v. 4, n. 4, 2019.

JUNIOR, William Lauer. **O manifesto ágil e uma breve introdução ao SCRUM**. São Vicente, SP: Ebook acessado do Kindle, 2020. 109 p.

MACHADO, Felipe Nery Rodrigues. **Análise e Gestão de Requisitos de Software Onde nascem os sistemas**. Saraiva Educação SA, 2018.

MACHADO, Marcos; MEDINA, Sérgio Gustavo. *SCRUM–Método Ágil: uma mudança cultural na Gestão de Projetos de Desenvolvimento de Software*. **Revista Científica Intraciência, Faculdade do Guarujá–UNIEESP**, v. 1, n. 1, p. 58-71, 2009.

MAZZOLA, Vitório. **Engenharia de software: Conceitos básicos**. Florianópolis: Editora da UFSC, c, v. 1, p. 140, 2006.

MIZRAHI, Saul Eliahú; CICERO, Janete Rocha. A tecnologia assistiva para promoção da aprendizagem e inclusão social do aluno com deficiência. **Revista Benjamin Constant, edição especial**, p. 54-70, 2016.

PASSERINO, L. M.; MONTARDO, S. P. **Inclusão social via acessibilidade digital: proposta de inclusão digital para Pessoas com Necessidades Especiais**. E-Compós, v. 8, 26 jun. 2007.

PRESSMAN, Roger; MAXIM, Bruce. **Engenharia de Software-8ª Edição**. McGraw Hill Brasil, 2016.

REINALDI, Leticia Ramos; DE CAMARGO JÚNIOR, Cláudio Rosa; CALAZANS, Angelica Toffano Seidel. Acessibilidade para pessoas com deficiência visual como fator de inclusão digital. **Universitas: Gestão e TI**, v. 1, n. 2, 2011.

SANTOS, José Paulo et al. TECNOLOGIA ASSISTIVA: um estudo sobre o uso de aplicativos para deficientes visuais. **Brasil Para Todos-Revista Internacional**, v. 4, n. 1, p. 59-69, 2017.

SILVA, Janaina; BRAGA, Juliana Cristina; DAMACENO, Rafael. **Estudo de aplicativos móveis para deficientes visuais no âmbito acadêmico**. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2015.

SILVA, Leandro Luquetti B. da; PIRES, Daniel Facciolo; NETO, Silvio Carvalho. **Desenvolvimento de aplicações para dispositivos móveis: tipos e exemplo de aplicação na plataforma IOS**. Franca/SP, 2015.

SIMAS, Victor L.; BORGES, Olimar T.; COUTO, Júlia M C.; et al. **Desenvolvimento para dispositivos móveis - Volume 2**. Porto Alegre: Grupo A, 2019.

SOMMERVILLE, Ian. **Engenharia de Software. 9o Edição**. PEARSON EDUCATION - BR, 2011.

---

**The SCRUM Framework Poster.** Disponível em:  
<<https://www.SCRUM.org/resources/SCRUM-framework-poster>>. Acesso em: 24 fev. 2021.  
VELOSO, Renato dos S. **Tecnologia da informação e comunicação, 1ª edição.** São Paulo:  
Editora Saraiva, 2012.