

Estudo de caso: avaliação da qualidade da especificação de requisitos para o desenvolvimento de um aplicativo educacional

Simone Borges Simão Monteiro (Universidade de Brasília)

**Giovanna Lourenço Rodrigues (Universidade de
Brasília)**

Pedro Griesinger Peres (Universidade de Brasília)

Pedro Pimentel Foizer (Universidade de Brasília)

Pedro Ponciano Lopes (Universidade de Brasília)

O artigo propõe demonstrar a importância na avaliação de requisitos de um software. Dentro de todas as etapas do desenvolvimento de um software, o planejamento é a etapa mais crucial para a sua construção. Nela, será feita a comunicação entre o planejador e o programador, sendo qualquer falha crucial para o produto final. Diante disso, utilizamos uma metodologia voltada para a aplicação de checklist, embasada em pesquisas bibliográficas, e validação dos requisitos de especificação do software levantados. Com isso, foi possível estruturar indicadores que mostraram os módulos dos requisitos com maior quantidade de erros e ponderações feitas. Por meio desse processo de qualidade, o planejamento de desenvolvimento do software se mostrou bem mais assertivo, mitigando custos financeiros e de tempo.

Palavras-chave: Garantia da Qualidade, ed-tech, ferramentas de qualidade, especificação de requisitos, avaliação de requisitos, checklist, aplicativo, mobile education e protótipo .

1. Introdução

1.1 Importância da Gestão da Qualidade no desenvolvimento de um software

Cometer erros é uma característica humana, sendo em grande parte das vezes, pequenos mas que podem ser muito custosos. Em meio a um cenário de desenvolvimentos de diversos softwares, a qualidade final do produto torna-se muito importante para se destacar em um mercado acirrado.

A qualidade de um software, como é descrito por Ava Franklin, "refere-se à capacidade de um produto de software de executar suas funções esperadas de maneira segura e sem falhas" (GOODCORE; 2020). Através do Quality Assurance (QA) é o processo de elencar erros no código do software que impactam na experiência do consumidor final e resolvê-los. Esse processo é dividido em duas etapas: testar e controle da qualidade. Basicamente, o primeiro se refere a parte prática em que será testada as diversas funcionalidades de um software visando encontrar falhas e consertar o produto. Já o controle da qualidade é direcionado a um produto pré finalizado, visando entender se os requisitos e especificações foram atendidos. Por fim, o QA busca unir os dois processos anteriores e dentro de todo o ciclo de qualidade do projeto chegou ao fim.



Fonte: Adaptado de FRANKLIN, Ava (GOODCORE, 2020)

Como em qualquer outro produto, a qualidade é também importante para os softwares. Seja para um sistema interno para poucos usuários quanto para um sistema complexo feito para milhões de pessoas. A Quality Assurance busca identificar erros visando diminuir perdas de

tempo em dinheiro, garantindo que o produto final seja seguro, competitivo e execute as funções sem falhas inesperadas.

2. Revisão bibliográfica

2.1. Ed-Tech (Educação por meio da Tecnologia)

A pandemia do coronavírus que está assolando o mundo desde 2020, acelerou ainda mais o aumento da demanda por educação como a tecnologia como o seu principal vetor (Credit Suisse Group; 2020). Por isso, cada vez mais está sendo revolucionado a forma de ensinar no mundo, mesmo sendo um dos setores que aceitou essas evoluções tecnológicas de forma mais lenta (Credit Suisse Group; 2020).

Com a onda de lockdowns no começo de 2020 como forma de contenção da pandemia da covid-19, mais de 90% de todos os alunos de escolas e faculdades estavam estudando de forma remota, em Abril de 2020 mais de 1,7 bilhões de alunos estavam estudando de forma remota (Credit Suisse Group; 2020).

Isso fez com que empresas que possuem o estudo à distância tivessem um crescimento nesse período, com destaque à Estácio no Brasil que cresceu 55% no primeiro quadrimestre de 2020 e da Afya Educational no Brasil que lançou cursos de treinamentos a residentes de medicina para melhor preparo ao combate ao coronavírus (Credit Suisse Group; 2020).

Deve ser ressaltado a importância de que a tecnologia deve se adequar ao ambiente em que for utilizada, todo o contexto e realidade deve ser mensurado para que a implementação da tecnologia proposta seja correta e efetiva, visando evitar problemas futuros (Bryant, J.; 2021).

A adaptação é essencial para a implementação, não pode ser algo muito disruptivo pois o impacto tecnológico pode ser o contrário do almejado, por isso deve-se sempre adequar a tecnologia ao ambiente educacional proposto, e não o contrário.

Nota-se que a tecnologia no ensino é um caminho sem volta e que deve se desenvolver cada vez mais (Credit Suisse Group; 2020).. O Brasil pode ser um grande polo nessa evolução pois é um dos países que mais tem potencial de crescimento com esse tipo de tecnologia.

2.2. Levantamento de requisitos

O levantamento de requisitos consiste, basicamente, em o que deve conter no software: fatores importantes a serem considerados na hora de levantar requisitos, como por exemplo linguagem, necessidades do cliente, usabilidade etc. (Davis, A.; 1993)

Requisitos na engenharia são difíceis de se levantar, não adianta simplesmente escrever o que o cliente diz que quer. Os requisitos são mutáveis e são alterados na medida em que mudanças ocorrem no processo. Assim, é de suma importância ter bons requisitos e o entender a dinâmica da natureza do processo, sendo o mais assertivo possível ao mesmo tempo que se é flexível. E é com esse pensamento que a pesquisa será levada, tentando sempre se adaptar às

necessidades dos clientes e alterando possíveis requerimentos que não agreguem ao projeto (Davis, A.; 1993)

Para a construção de qualquer software é necessário o estabelecimento e mensuração de requisitos para mensuração da qualidade do software projetado. Assim, no estabelecimento de métricas é importante pois ele que irá medir os atributos processo do software, os intermediários e, também, o resultado final gerado pelo software, isso só consegue ser alcançado com o levantamento de requerimentos no desenvolvimento do software e de suas especificações (Davis, A.; 1993)

Os requisitos específicos de um software (SRS) é um documento que descreve todas as características externas analisáveis de comportamentos e características esperadas por um sistema de software. Isso mostra o quão importante é a análise da qualidade das entregas sobre os requerimentos do software para que os programadores possam com clareza executar as atividades almejadas pelos stakeholders e público-alvo (Davis, A.; 1993)

Um SRS de qualidade consegue rastrear uma boa parte dos erros cometidos no desenvolvimento do software, desde seus requerimentos até detalhes de design. Assim, quando mais se consegue detectar e mensurar a qualidade dos SRS, maior a facilidade de detecção de erros no SRS, assim a equipe de desenvolvedores estará melhor treinada para a antecipação de problemas que possa haver no software. Tudo isso gera uma eficiência de custos e evita retrabalhos e maior alinhamento da equipe acerca do que será requerido pelo software (Davis, A.; 1993)

Com SRS, a fase de requisitos é primordial para detecção de possíveis erros na plataforma. Assim, o trabalho desenvolvido deve assegurar pela qualidade das entregas dos requerimentos levantados pois eles são um grande termômetro para a avaliação/comportamento da plataforma na maioria de formas testadas (Davis, A.; 1993). Além de gerar resultados eficientes e menos erros com a plataforma já em uso.

Os requisitos não precisam ser longos ou caros. Eles devem ser práticos a fim de que possam realmente expressar e documentar o que quais que serão utilizados, eles são essenciais a qualquer projeto para ter melhores especificações do que será feito no projeto (Davis, A.; 1993).

Será necessário no projeto uma análise quais os requisitos que são realmente necessários aos consumidores, é nesse ponto que pode nascer a inovação. Muitas vezes os clientes já possuem uma visão mais enviesada do output gerado pelo projeto, entretanto, com uma percepção a fundo de quais seriam as reais demandas do cliente é assim que pode gerar cada vez mais valor a sociedade e, principalmente, aos stakeholders (Duffey, C. K., & Stratford, R. P.;

1989).

Dessa forma, foram elencados alguns requisitos e indicadores essenciais que devem estar presentes no SRS analisado pela pesquisa:

- Natureza: São listadas algumas considerações que os responsáveis por escrever a ERS devem ter no que tange a funcionalidade do software, as interfaces externas, a performance, os atributos e as restrições de design imposta na implementação (IEEE Std 830-1998; 1998).
- Ambiente: Entender a atuação do ERS no plano de todo o projecto, que é definido no documento IEEE Std 610.12-1990. O software pode conter essencialmente toda a funcionalidade do projeto ou pode ser parte integrante de um sistema maior. Neste último caso, tipicamente existe um documento de ErS que documenta as interfaces entre o sistema e esta parte do software, impondo assim exigências externas de performance e funcionalidade do software. Sendo que a ERS deve estar em concordância e expandir estas exigências de sistema (IEEE Std 830-1998; 1998).
- Características de um bom SRS: correto, não ambíguo, completo, consistente, elencado por importância, verificável, modificável e rastreável (IEEE Std 830-1998; 1998).
- Preparação Conjunta: tanto o consumidor quanto o fornecedor devem escrever a SRS de forma conjunta (IEEE Std 830-1998; 1998).
- Evolução: durante o desenvolvimento do software são levantados questionamentos que não haviam sido feitos no início. Com isso, devem ser levantados todos os requerimentos, que tenham conhecimento no momento da elaboração do SRS (IEEE Std 830-1998; 1998).
- Prototipagem: o protótipo faz com que o consumidor consiga, de forma mais prática, dar feedbacks do que lendo o SRS. Ele produz novas respostas e novas perguntas também (IEEE Std 830-1998; 1998).
- Integrar um designer: o escritor do SRS deve definir claramente as restrições necessárias e projetar um design específico (IEEE Std 830-1998; 1998).

Esse é um dos principais objetivos que o levantamento de requisitos gerados para o "Mobile Education" é tão importante, pois tentará encontrar reais requisitos que são almejados pelos stakeholders a fim de que a plataforma possa aumentar cada vez mais o seu impacto na sociedade (Duffey, C. K., & Stratford, R. P.; 1999).

2.3. Metodologia para validação de requisitos

Essa metodologia está descrita no artigo “IEEE Recommended Practice for Software Requirements Specifications”. (IEEE Std 830-1998; 1998)

O conteúdo desse documento foi elaborado pelo Grupo de Trabalho de Harmonização de Dados do Ciclo de Vida do Comitê de Padrões de Engenharia de Software da IEEE Computer Society. (IEEE Std 830-1998; 1998)

O panorama geral do documento é:

- a) Escopo;
- b) Lista de publicações que devem ser consultadas junto ao documento de prática recomendada;
- c) Definições;
- d) Considerações para produzir uma boa Especificação de Requisitos de Software (ERS);
- e) Componentes essenciais da especificação de requisitos de software.

Esta prática recomendada também tem dois anexos, um que fornece modelos de formato alternativo e um que fornece diretrizes para conformidade com a publicação IEEE / EIA 12207.1-1997.(IEEE Std 830-1998)

Verificou-se que a metodologia adotada como referência para a elaboração de ferramentas de avaliação dos requisitos desta pesquisa é amplamente utilizada em pesquisas, estudos e desenvolvimento de produtos relacionados a desenvolvimento de software. Assim, escolheram-se três artigos de publicações independentes e dois de publicações vinculadas à própria IEEE para serem comentados.

O artigo “A Kinect-Based System for Lower Limb Rehabilitation in Parkinson’s Disease Patients: a Pilot Study” fala de técnicas emergentes de realidade virtual e de interfaces de usuário naturais para oferecer novas possibilidades no campo da reabilitação. Os pesquisadores projetaram um jogo de reabilitação baseado em um dispositivo de baixo custo (Microsoft KinectTM) conectado a um computador pessoal. Esse jogo fornece aos pacientes com doença de Parkinson (DP) uma forma motivadora de realizar vários exercícios de reabilitação motora para melhorar sua reabilitação (PALACIOS-NAVARRO et al., 2015)

O segundo artigo selecionado, intitulado “From requirements to UML models and back: how automatic processing of text can support requirements engineering”, pretende mostrar que é possível acelerar significativamente o processo de criação de textos com menos falhas. A teoria utilizada como referência fala como a engenharia de software deve ser um processo estruturado, o que inclui a criação da especificação textual do sistema previsto, bem como a

criação de modelos para os engenheiros de software e como é comum ocorrer uma certa deterioração durante o processo devido a mudanças de requisitos, decisões de implementação, etc. Assim, a sincronização de requisitos textuais, modelos e o software real são indispensáveis para entregar o software conforme especificado. Como o custo de manter a consistência manualmente é muito alto, a equipe utiliza um sistema de feedback de engenharia de requisitos próprio para automatizar o processo de manter a especificação textual e os modelos consistentes quando os modelos mudam. Para melhorar o processamento geral das especificações de linguagem natural, nossa abordagem encontra falhas nas especificações de linguagem natural (Landhäußer et al., 2013)

A publicação “A method for assessing confidence in requirements analysis” tem como objetivo descrever um método novo, simples e prático para avaliar a confiança dos pesquisadores em um conjunto de requisitos. O contexto é a fase de desenvolvimento de um software em que os gerentes, analistas e designers geralmente precisam saber se o trabalho de análise de requisitos foi feito de forma suficiente e se é ou não seguro para prosseguir para o estágio de design. (BONESS et al., 2011)

“Guidelines for Eliciting Usability Functionalities” foi a quarta publicação escolhida. Nesse documento é proposto uma abordagem baseada no desenvolvimento de diretrizes específicas que capitalizam elementos-chave que intervêm no processo de elicitação e especificação de recursos de usabilidade. As diretrizes fornecem aos analistas de requisitos um repositório de conhecimento que pode ser utilizado para fazerem-se as perguntas certas e capturarem-se informações precisas sobre os requisitos de usabilidade. A fundamentação teórica do artigo fala da relevância da usabilidade e de seu entendimento pelas partes envolvidas na elicitação de requisitos, já que, como todo atributo da qualidade, ela impõe restrições específicas aos componentes de software. Idealmente, os recursos que aumentam a usabilidade do sistema de software devem ser considerados desde os primeiros estágios de desenvolvimento. Contudo, o que infere-se da análise dos casos reais é que a tarefa de descobrir e documentar recursos de usabilidade requer um conhecimento de usabilidade muito acima do da maioria dos engenheiros de requisitos, desenvolvedores e usuários. (JURISTO et al., 2007)

Por fim, o último artigo analisado foi a de título “Identifying, Assigning, and Quantifying Crosscutting Concerns”. Ele propõe uma técnica para localizar e apurar o nível de questões transversais. (EADDY et al., 2007)

3. Metodologia

3.1. Método de pesquisa

Este trabalho começou com pesquisas realizadas nos sites Web of Science, no Google Acadêmico e na base de artigos do IEEE, utilizando as palavras chaves relacionadas ao objetivo desta pesquisa, palavras chaves como, software, requirements, mobile, education, requirements specifications and quality assurance, visto que essas palavras são relacionadas para o desenvolvimento de um software com a qualidade pretendida. Dessa forma o principal artigo usado como base para este trabalho foi o IEEE Recommended Practice for Software Requirements Specifications (IEEE), um artigo que foi elaborado pelo Grupo de Trabalho de Harmonização de Dados do Ciclo de Vida do Comitê de Padrões de Engenharia de Software da IEEE Computer Society.

Realizada a pesquisa e selecionados os principais artigos a serem usados como base, foi feito um estudo para realizar um checklist que assegure as qualidades dos requisitos, agregando valor aos requisitos levantados pelo grupo de PSP2. Por meio do estudo dos artigos, em especial o artigo IEEE Recommended Practice for Software Requirements Specifications (IEEE), o checklist foi desenvolvido, estabelecendo critérios de qualidade abordados pelo artigo, objetivando avaliar as entregas e validá-las seguindo o cronograma de entrega da equipe de levantamento de requisitos.

Com o checklist desenvolvido, foram realizadas reuniões com a equipe de levantamento de requisitos, para fazer um alinhamento das entregas e apresentar o checklist de qualidade, assim como, os feedbacks dessas entregas, em relação ao checklist. Esses feedbacks tem por objetivo que os requisitos entregues sigam os critérios de qualidade desenvolvidos, validando as entregas e dando continuidade a pesquisa, as entregas que não se adequarem ao checklist seriam corrigidas e entregues novamente para uma nova validação, até que a qualidade almejada fosse obtida, recebendo uma última validação.

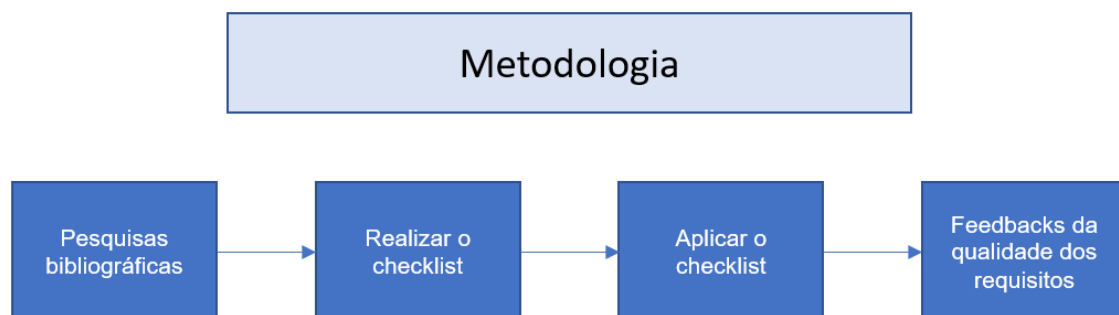
Frente aos fatos apresentados, nossa pesquisa foi baseada no método de pesquisa exploratória quanto aos objetivos, pois envolve um levantamento bibliográfico e um aprofundamento do tema com pesquisas artigos elaborados por pessoas experientes ao objeto de pesquisa (GIL,2008), já no que concerne o procedimento técnico da pesquisa ela se enquadra em um estudo de caso focado no objeto *Mobile Education*, permitindo um amplo e detalhado conhecimento do tema. (GIL,2008). Com essas bases de pesquisas estabelecidas, nossa metodologia foi estruturada em estruturação da pesquisa e método de pesquisa.

3.2. Estruturação da pesquisa

A pesquisa é uma colaboração entre duas disciplinas com equipes distintas do curso de Engenharia de Produção da Universidade de Brasília. Com isso, se tratando de um desenvolvimento de software voltado para a educação mobile, este artigo foca na parte da garantia de qualidade dos requisitos, estabelecidas pela outra equipe. Sendo assim, a equipe da disciplina de Projeto de Sistemas de Produção 2 (PSP2) fez o levantamento de requisitos para desenvolver um protótipo de um aplicativo e a equipe da disciplina de Projeto de Sistemas de Produção 5 (PSP5), foco do artigo, realizou um checklist para assegurar a qualidade desses requisitos.

O controle das etapas dessa pesquisa foi realizado por meio da ferramenta Trello, tendo por lá o controle de todas as entregas, etapas e arquivos usados para se realizar esse trabalho. Além disso as entregas foram estabelecidas por meio de sprints, seguindo um cronograma pré-estabelecido. Abaixo tem-se a Figura 2 resumindo o processo de estruturação da pesquisa.

Figura 2 – Estruturação da pesquisa



Fonte: Autores (2021)

4. Resultados e discussões

Os resultados apresentados aqui foram obtidos por meio de uma reunião explicativa do artigo IEEE Recommended Practice for Software Requirements Specifications (IEEE) com o grupo de levantamentos de requisitos da matéria de Projeto de Sistemas de Produção 2, nessa pesquisa é nomeado como PSP2 para facilitar a compreensão, da Universidade de Brasília. Além disso, foi elaborado um checklist para garantir a qualidade do projeto.

4.1. Apresentação e discussão da metodologia usada para levantar requisitos

A metodologia apresentada no artigo IEEE Recommended Practice for Software Requirements Specifications, foi utilizada como base para a elaboração do checklist de qualidade dos requisitos, antes da elaboração do checklist, foi preparada uma apresentação onde foi discutida a importância do levantamento de requisitos e de sua qualidade para o sucesso da implementação final do software de mobile education.

O objetivo dessa discussão foi apresentar o método que será utilizado para avaliar os requisitos, bem como auxiliar e mostrar os pontos que serão avaliados e sua importância final para o desenvolvimento do aplicativo mobile education.

Com a discussão foi alinhado a forma de validação dos requisitos seguindo a metodologia abordada, aumentando o entendimento e direcionando o grupo de levantamento de requisitos a alcançar a qualidade almejada. Mais a frente do artigo será apresentado o resultado final dessa validação.

4.2. Checklist de Requisitos

Após a leitura do artigo da IEEE, um dos resultados chave para a equipe na pesquisa é a avaliação dos requisitos levantados pela a equipe da Universidade de Brasília com os catadores do lixo da Estrutural.

O principal resultado produzido pelo grupo da pesquisa é a avaliação da primeira parte da montagem do software do Mobile Education, que são os levantamentos de requisitos e análise do design do protótipo do software que será desenvolvido pela Universidade Aalborg.

Por isso que foi importante a apresentação do artigo da IEEE a fim de que esses requisitos sejam mensuráveis e utilizáveis baseados nos principais atributos levantados e demandados por todas as partes envolvidas no processo.

O principal papel da equipe é a avaliação desses requisitos, criando indicadores que mensuram o desempenho no levantamento, a fim de que possam ser estabelecidos padrões de qualidade no levantamento de requisitos. Visando um estabelecimento de um padrão no estabelecimento desses requisitos levantados pelo grupo de PSP2.

Assim a equipe montou um checklist com mensurando os requisitos levantados pelo grupo de PSP2 após as entrevistas com o possível público-alvo do aplicativo do Mobile Education.

Critério	Descrição do Item
Não Ambiguidade	Cada requisito expresso apresenta apenas uma interpretação.
Completeness	Todos os requisitos relevantes, desde os relacionados a funcionalidade, performance, restrição de design, atributos ou interfaces externas, são identificados e especificados no documento.
Completeness	Há uma definição das respostas do software para todas as classes de dados de entrada realizáveis em todas as classes de situações realizáveis.
Completeness	Todas as figuras, tabelas e diagramas foram devidamente identificadas e referenciadas, bem como todos os termos e unidades de medida foram definidos.
Consistência	A descrições dos requisitos levantados não são conflitantes ou se contradizem em relação a características dos objetos do mundo real, lógica de ações, cronologia de ações ou inconsistência no uso de termos que descrevem um mesmo objeto.
Classificado por importância e/ou estabilidade	Todos os requerimentos levantados possuem uma identificação de nível de importância, estabilidade ou necessidade.
Verificabilidade	A descrição de todos os requisitos levantados apresenta termos concretos, termos mensuráveis ou valores quantitativos.
Verificabilidade	O processos de verificação do requisitos levantado é finito.
Verificabilidade	Os processo de verificação do requisito levantado é economicamente viável.
Verificabilidade	O processo de verificação não promove respostas subjetivas.
Modificáveis	Os requisitos são passíveis de modificações nas diferentes etapas do processo de desenvolvimento do produto de software.
Modificáveis	As mudanças são fáceis de incorporar ao documento.
Modificáveis	Cada requisito é expresso e explicado separadamente.
Rastreável	A origem dos requisito é clara e referenciada no documento. Cada requisito referencia explicitamente sua fonte em documentos anteriores.
Rastreável	Existem documentos adjacentes que expliquem todas mudanças incorporadas na especificação de requisitos e a forma como foi feito o levantamento de requisitos.
Rastreável	Cada requisito do documento tem um nome ou número de referência exclusivo.

Fonte: Autores (2020)

Para mensurar e efetuar o checklist dos requisitos, foi preenchido alguns campos que posteriormente seriam utilizados na mensuração dos indicadores gerados após a execução do checklist.

Primeiramente, foi preenchido a sua avaliação como Adequado, Inadequado ou Não Se Aplica, depois o Grau de importância como Alta, Média, Baixa. O resultado, a quantidade e as observações acerca do requisito mensurado. Um exemplo do preenchimento do *checklist* é apresentadona Figura 4.

Figura 4 – Preenchimento do *checklist* de requisitos

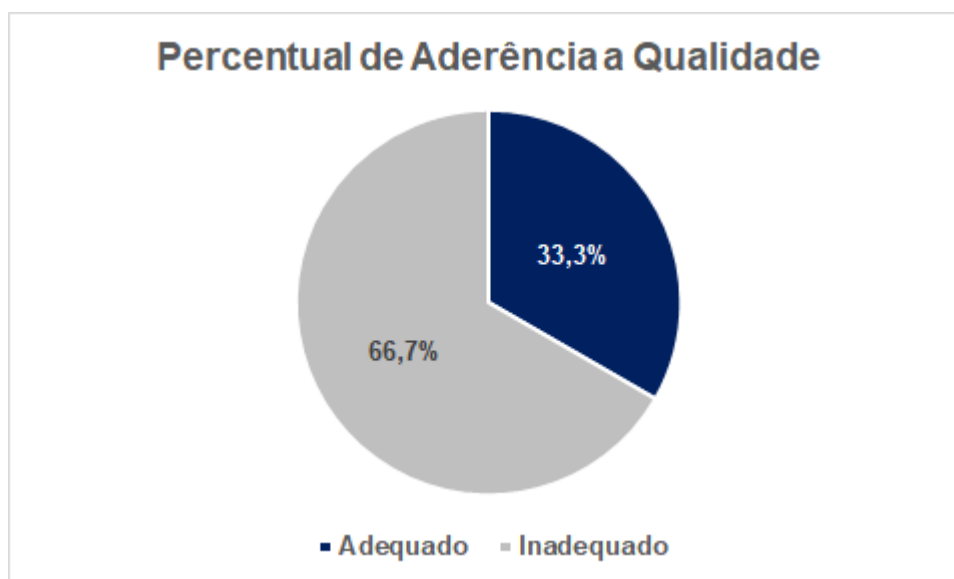
Item	Critério	Descrição do Item	Avaliação			Grau de Importância	Resultado	Quantidade	Observação
			Adequado	Inadequado	Não se Aplica				
1	Não Ambiguidade	Cada requisito expresso apresenta apenas uma interpretação.		X		Alta		6	Dentro de todos os módulos de requisito foi expresso o sub-item "description/purpose". A escolha de duas palavras que

Fonte: Autores (2021)

4.3 Indicadores da qualidade dos requisitos

Com o checklist já efetuado foram mensurados alguns indicadores acerca dos requisitos levantados pela a equipe de PSP2. Primeiramente, foi observado a representatividade aderência a qualidade dos requisitos levantados, nota-se que apenas 33,3% dos requisitos levantados estavam adequados na sua aderência a qualidade, como pode-se ver na Figura 5 a seguir.

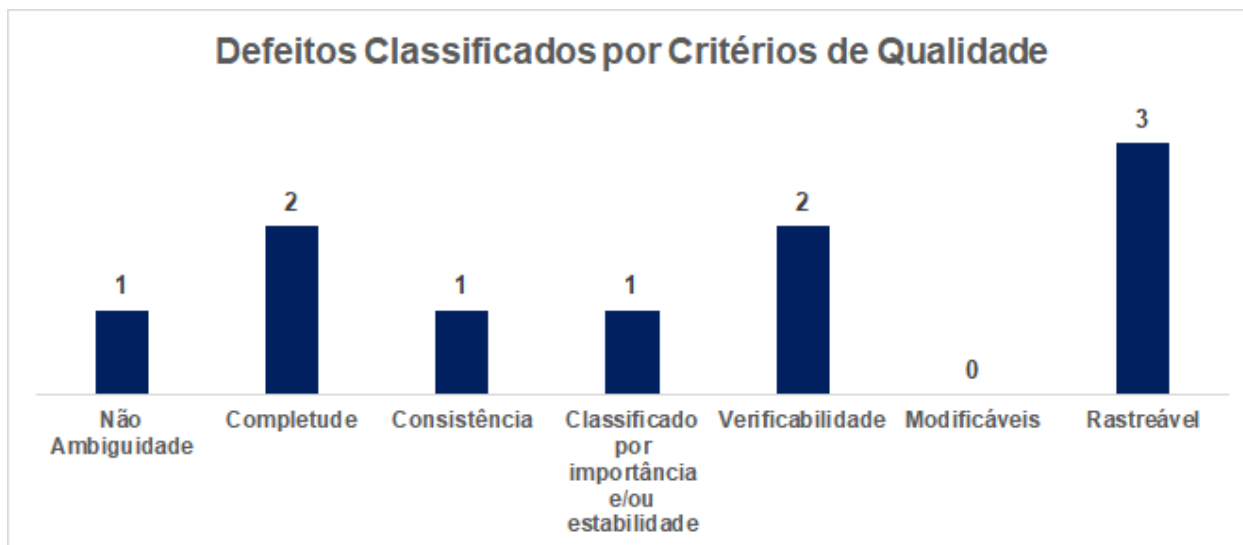
Figura 5: Percentual de Aderência a Qualidade



Fonte: Autores (2020)

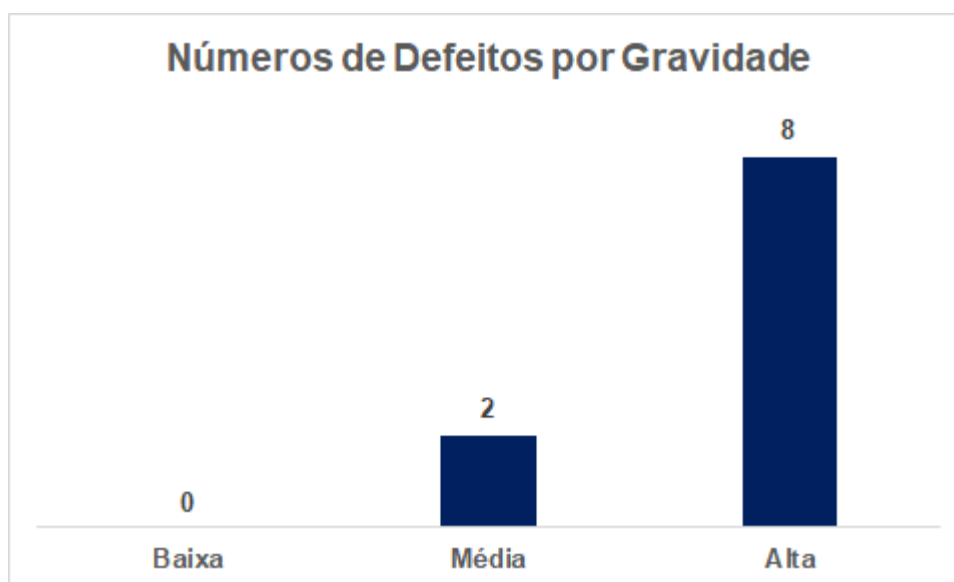
Por escassez de tempo, ocorreu apenas uma validação dos requisitos levantados assim foi analisado quais foram os critérios e o grau de importância dos requisitos levantados em que ocorreram falhas:

Figura 6: Defeitos Classificados por Critérios de Qualidade dos Requisitos do Software



Fonte: Autores (2021)

Figura 7: Número de Defeitos por Gravidade dos Requisitos do Software



Fonte: Autores (2021)

Percebe-se com base nas figuras, que dos dez requisitos que foram classificados como inadequados, na qual a maioria dessas falhas concentravam-se na rastreabilidade, completude e verificabilidade dos requisitos listados, concentrando-se nesses 3 critérios 70% das falhas registradas no checklist.

Além disso, também, foi mensurado a gravidade de cada um dos requisitos listados pelo seu grau de importância, assim 80% dos requisitos inadequados eram de alta importância demonstrando que algumas mudanças são essenciais para que o documento possa estar melhor adequado para o melhor resultado na construção do aplicativo do Mobile Education.

4.4 Checklist do Protótipo

Pelo alto contato com o possível público-alvo, a equipe de PSP2 construiu um protótipo da interface do aplicativo do *Mobile Education* visando ajudar os desenvolvedores do aplicativo a atenderem demandas essenciais para que o produto possa gerar o maior valor possível.

Com isso, um checklist do protótipo foi desenvolvido pela equipe da pesquisa na qual iria mensurar a qualidade do produto desenvolvido pela equipe PSP2.

Assim, alguns critérios foram levantados para avaliar o protótipo, com um design bastante similar ao primeiro checklist montado pela equipe:

Figura 8: Descrição dos Itens do Checklist do Protótipo

Usabilidade	O protótipo é aplicável ao mundo real? Ele está integrado a realidade dos usuários?
Usabilidade	O design possui um pontos de prevenção e recuperação de erros?
Usabilidade	Todos os dados inseridos resultam nas saídas desejadas do protótipo?
Usabilidade	Todas as opções interativas do protótipo possuem função real?
Usabilidade	O protótipo antecipa de forma clara e intuitiva os comportamentos do sistema?
Conformidade	Existe algum recurso de suporte no aplicativo?
Conformidade	Todas as funcionalidades preenchem uma ordem de usabilidade lógica?
Conformidade	Todos os cenários possíveis que podem acontecer ao usar o aplicativos são apresentados no wireframe?
Experiência do Usuário	A paleta de cores permite a visualização de todos os componentes do protótipo?
Usabilidade	É possível fazer buscas de aulas no aplicativo?
Experiência do Usuário	A paleta de cores permite o direcionamento para os campos de utilização?
Experiência do Usuário	Em um primeiro contato, o usuário é apresentado aos recursos essenciais para utilizar o aplicativo ?
Experiência do Usuário	É possível fazer buscas de tópico no fórum de perguntas?

Fonte: Autores (2021)

Para mensurar e efetuar o checklist do protótipo, foi preenchido alguns campos da mesma forma em que foi realizado no primeiro checklist desenvolvido pela equipe. Classificando os itens em três grupos, Conformidade, Experiência do Usuário e Usabilidade.

Primeiramente, foi preenchido a sua avaliação como Adequado, Inadequado ou Não Se Aplica, depois o Grau de importância como Alta, Média, Baixa. O resultado, a quantidade e as observações acerca do requisito mensurado.

Figura 9: Preenchimento do Checklist do Protótipo

Critério	Descrição do Item	Avaliação			Grau de Importância	Resultado	Quantidade	Observação
		Adequado	Inadequado	Não se Aplica				
Usabilidade	O protótipo é aplicável ao mundo real? Ele está integrado a realidade dos usu	x			Alta	Adequado		

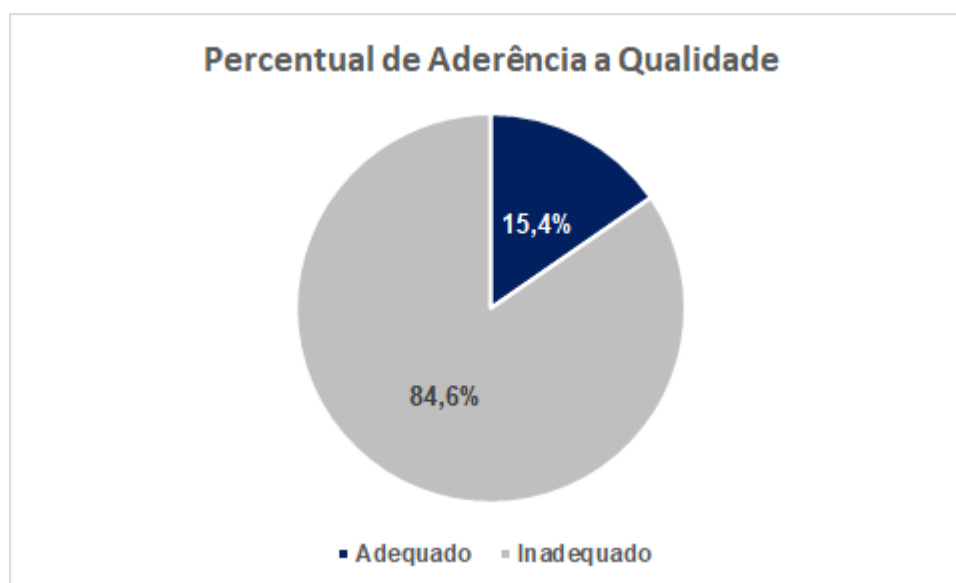
Fonte: Autores (2021)

4.5 Indicadores do Protótipo

Após a avaliação dos itens, alguns indicadores da qualidade foram levantados para avaliação do

protótipo. Primeiramente, foi avaliado a representatividade da aderência a qualidade dos indicadores levantados, notando que apenas 15,4% dos indicadores levantados no checklist estavam adequados:

Figura 10: Percentual de Aderência a Qualidade

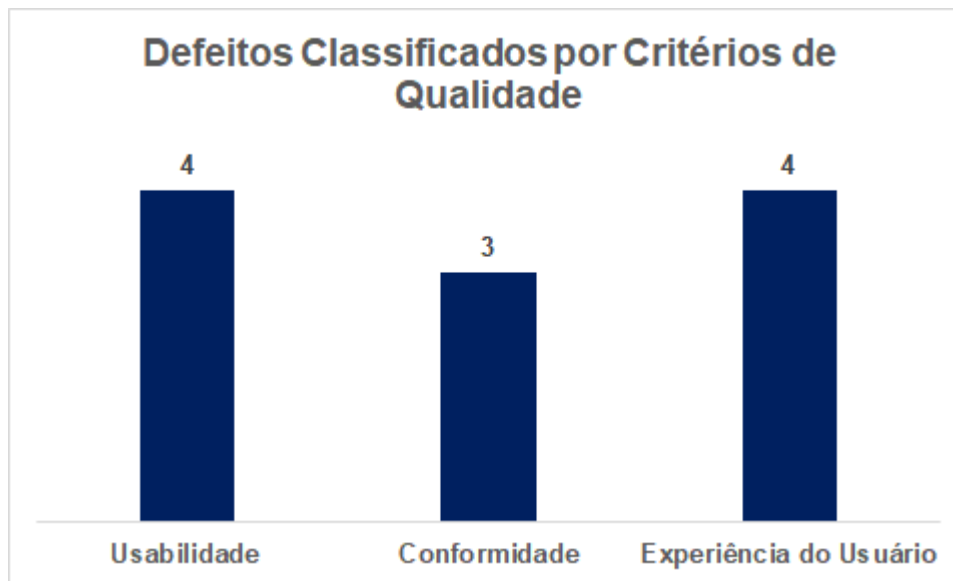


Fonte: Autores (2021)

Pela escassez do tempo, só ocorreu uma validação dos itens gerados pela equipe de PSP2. Assim foi destacado as falhas para que possam ser corrigidas e melhor aproveitadas pela equipe do Mobile Education.

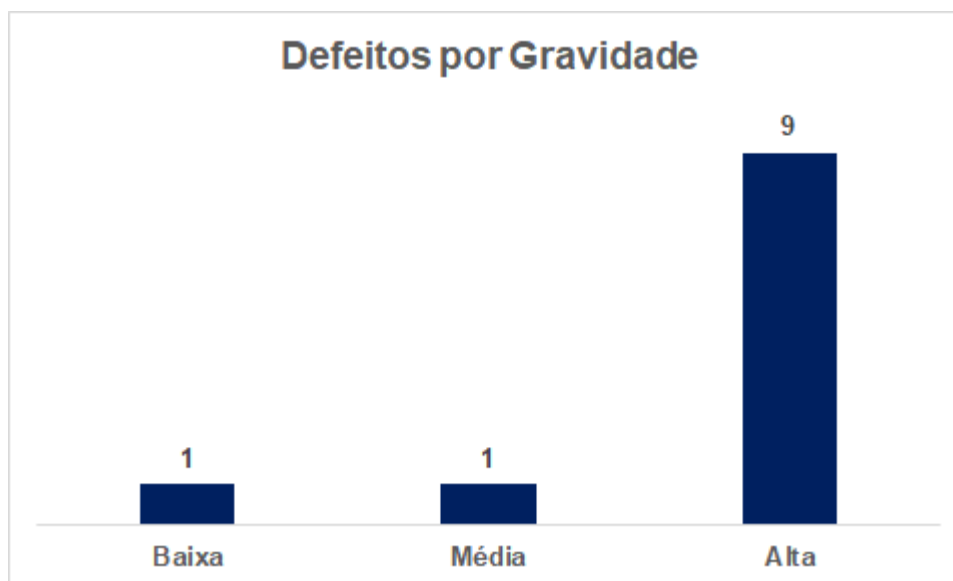
Por isso, foi analisado quais foram os critérios e o grau de importância dos requisitos levantados em que ocorreram falhas:

Figura 11: Dados Classificados por Critérios de Qualidade do Protótipo do Aplicativo



Fonte: Autores (2021)

Figura 12: Defeitos por Gravidade do Protótipo do Aplicativo do *Mobile Education*



Fonte: Autores (2021)

Percebe-se que dos onze indicadores que foram classificados como inadequados, oito estão elencados nos critérios de Experiência do Usuário e Usabilidade, representando 72,3% das falhas mensuradas no checklist. Demonstrando que há vários pontos de melhora no protótipo do aplicativo do Mobile Education.

Outro ponto de importante avaliação, é a gravidade dos indicadores inadequados. Assim percebe-se na figura 12, que 81,8% dos itens medidos possuem alta gravidade o que mostra que

mudanças são necessárias para que a construção dessa interface seja a melhor possível.

5. Considerações Finais

Mesmo não tendo uma etapa de segunda verificação dos checklists, percebe-se como essa ferramenta é muito importante para a detecção de erros e onde é ponto de maior dificuldade das equipes que realizam essa atividade para que no futuro possam melhorar cada vez mais o levantamento dos requisitos e a construção da interface.

Com isso, deve-se entender que a atualização do checklist e a validação constante dos pontos levantados por cada um deles pois percebe-se que a maioria das falhas geradas possui um Alto Grau de Importância, 80% no checklist dos requisitos levantados e 81,8% da interface do protótipo.

Além disso, as ferramentas utilizadas para mensuração da qualidade, como o checklist utilizado na pesquisa, e seus indicadores gerados são uma ótima forma para avaliação de métodos e dos processos realizados, pois ele serve como uma forma constante de corrigir falhas e aprimorar os processos beneficiando todos os stakeholders. Podendo, assim, gerar melhores resultados no output.

Nota-se, também, o fato do projeto do Mobile Education ser um projeto global e que depende de vários stakeholders de diferentes países e de diferentes culturas. É essencial que a comunicação entre as partes envolvidas deve estar extremamente alinhada visando evitar problemas que impactam o andamento do projeto.

REFERÊNCIAS

BRYANT, J.; CHILD, F.; DORN, E.; HALL, S. New global data reveal education technology's impact on learning. **Mckinsey and Company**, 12 jun. 2020. Disponível em: <https://www.mckinsey.com/industries/public-and-social-sector/our-insights/new-global-data-reveal-education-technologys-impact-on-learning>. Acesso em: 04 out. 2021.

CREDIT SUISSE GROUP AG and/or its affiliates. Educational technology: Coronavirus and beyond. **Credit Suisse**, 08 abr. 2020. Disponível em: <https://www.credit-suisse.com/about-us-news/en/articles/news-and-expertise/education-technology-and-coronavirus-pandemic-202004.html>. Acesso em: 08 set. 2021.

DAVIS, Alan; et al. **Identifying and measuring quality in a software requirements specification**. Proceedings First International Software Metrics Symposium. Ieee, 1993.

FRANKLIN, Ava. Importance of Quality Assurance In Software Development. **GoodCore**, 2020. Disponível em: <https://www.goodcore.co.uk/blog/importance-of-quality-assurance/>. Acesso em: 08 de set. 2021.

IEEE Computer Society. Software Engineering Standards Committee, and IEEE-SA Standards Board. **IEEE Recommended Practice for Software Requirements Specifications**. Vol. 830. No. 1998. IEEE, 1998.

Kenneth Boness, Anthony Finkelstein, Rachel Harrison. **From a method for assessing confidence in requirements analysis, Information and Software Technology**. Volume 53, Issue 10, 1084-1096 (2011). <https://doi.org/10.1016/j.infsof.2011.05.003>.

KNAUSS, Eric. Assessing the Quality of Software Requirements Specifications. FG Software Engineering, Leibniz Universität Hannover – Welfengarten 1, 30167 Hannover, Germany

LANDHÄUBER, M., KÖRNER, S.J. & TICHY, W.F. **From requirements to UML models and back: how automatic processing of text can support requirements engineering**. Software Qual J 22, 121–149 (2014). <https://doi.org/10.1007/s11219-013-9210-6>

M. Eaddy, A. Aho and G. C. Murphy, **"Identifying, Assigning, and Quantifying Crosscutting Concerns"**, First International Workshop on Assessment of Contemporary Modularization Techniques (ACoM '07), 2007, pp. 2-2, doi: 10.1109/ACOM.2007.4.

N. Juristo, A. Moreno and M. Sanchez-Segura, **"Guidelines for Eliciting Usability Functionalities"**, in IEEE Transactions on Software Engineering, vol. 33, no. 11, pp. 744-758, Nov. 2007, doi: 10.1109/TSE.2007.70741.

PALACIOS-NAVARRO, G., GARCÍA-MAGARIÑO, I. & RAMOS-LORENTE, P. **A Kinect-Based System for Lower Limb Rehabilitation in Parkinson's Disease Patients: a Pilot Study**. J Med Syst 39, 103 (2015). <https://doi.org/10.1007/s10916-015-0289-0>

YOUNG, R.R. **The Requirements Engineering Handbook**. Editora Artech House, Norwood MA, 2004.