



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CAMPUS QUIXADÁ**  
**CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE**

**FRANCISCO ELENILSON PESSOA MONTEIRO**

**CLASSIFIQUI: UM JOGO SÉRIO PARA O ENSINO DE REQUISITOS DE  
SOFTWARE**

**QUIXADÁ**  
**2021**

FRANCISCO ELENILSON PESSOA MONTEIRO

CLASSIFIQUI: UM JOGO SÉRIO PARA O ENSINO DE REQUISITOS DE SOFTWARE

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Software.

Orientadora: Profa. Dra. Rainara Maia Carvalho

QUIXADÁ

2021

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

- M776c Monteiro, Francisco Elenilson Pessoa.  
Classifiqui : um jogo sério para o ensino de requisitos de software / Francisco Elenilson Pessoa  
Monteiro. – 2021.  
67 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá,  
Curso de Engenharia de Software, Quixadá, 2021.  
Orientação: Profa. Dra. Rainara Maia Carvalho.
1. Engenharia de Requisitos. 2. Jogos de Computador. 3. Ensino. I. Título.

CDD 005.1

---

FRANCISCO ELENILSON PESSOA MONTEIRO

CLASSIFIQUI: UM JOGO SÉRIO PARA O ENSINO DE REQUISITOS DE SOFTWARE

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Software.

Aprovada em: \_\_\_/\_\_\_/\_\_\_

BANCA EXAMINADORA

---

Profa. Dra. Rainara Maia Carvalho (Orientadora)  
Universidade Federal do Ceará (UFC)

---

Profa. Dra. Paulyne Matthews Jucá  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Enyo José Tavares Gonçalves  
Universidade Federal do Ceará (UFC)

Aos meus pais, Neuma e Edenilson. Mãe, por mais que a senhora não esteja mais entre nós em forma física sei que está em forma espiritual e sei que cuidou de mim durante essa caminhada. Minhas irmãs e cunhados, por sempre me apoiarem.

## **AGRADECIMENTOS**

A minha família, em especial meus pais, Maria Neuma Pessoa Monteiro e Ednilson Saraiva Monteiro e minhas irmãs Edineuma Fátima Pessoa Monteiro e Maria Elineuna Pessoa Monteiro pelo apoio, suporte, carinho e amor incondicional.

À Eduarda, Edval, Caio, Publio, Samuel e Larissa, pela amizade verdadeira e companheirismo durante toda a graduação e que dure a vida inteira.

Aos meus amigos pela amizade e apoio.

Aos participantes que se dispuseram a participar da coleta de dados.

A Profa. Dra. Rainara Maia Carvalho, por ter aceitado me guiar nesta caminhada, pela paciência e pela excelente orientação.

Aos professores participantes da banca examinadora Enyo José Tavares Gonçalves e Paulyne Matthews Jucá pelo tempo, pelas valiosas colaborações e sugestões. A todos que direta ou indiretamente contribuíram para a realização deste trabalho

“A persistência é o menor caminho do êxito.”

(Charles Chaplin)

## RESUMO

Com o advento das tecnologias e da Engenharia de Software (ES), a demanda pelo uso de produtos de software no mundo moderno em que vivemos cresceu. Uma das principais áreas da Engenharia de Software é a Engenharia de Requisitos (ER), pois atende às necessidades dos clientes que desejam e irão utilizar o produto de software. A maior parte do tempo de desenvolvimento é gasta nessa área, pois os requisitos requerem tempo para serem construídos e validados. Assim, qualquer problema nesta fase resultará em problemas para o resto do projeto. Uma das atividades da ER é a Análise de Requisitos, onde estes são classificados em diversos tipos, como Requisitos Funcionais, Regras de Negócio, entre outros. Erros nesta classificação podem causar problemas graves porque os desenvolvedores criarão especificações a partir desta classificação. Assim, surge a necessidade de apoiar o ensino desta classificação. Uma das abordagens que estão sendo utilizadas positivamente para o ensino de diferentes áreas do conhecimento é o uso de *serious games*. Nesse contexto, este trabalho propõe um jogo sério para auxiliar no ensino da classificação de requisitos de software. Para isso, foi desenvolvido um jogo digital chamado CLASSIFIQUI utilizando tecnologias como *React Native* e *Firebase*. Foi realizada uma avaliação com os alunos para coleta de dados que sobre o jogo proposto. A avaliação foi realizada com alunos da Universidade Federal do Ceará - Campus Quixadá dos cursos de Engenharia de Software e Sistemas de Informação. Os resultados obtidos indicam que o jogo CLASSIFIQUI pode ser útil para o ensino prático de classificação de requisitos.

**Palavras-chave:** Engenharia de Requisitos. Jogos de Computador. Ensino.

## ABSTRACT

With the advent of technologies and Software Engineering (SE), the demand for software products in the modern world we live in has grown. One of the main areas of Software Engineering is Requirements Engineering (RE), as it meets customers' needs to use the software product. Most of the development time is spent in this area, as the requirements require time to be built and validated. Thus, any problems at this stage will result in problems for the rest of the project. One of the RE's activities is the Requirements Analysis, where these are classified into several types, such as Functional Requirements, Business Rules, among others. Errors in this classification can cause severe problems because the developers will create specifications from this classification. Thus, there is a need to support the teaching of this classification. One of the approaches used positively for teaching different areas of knowledge is the use of *serious games*. In this context, this work proposes a serious game to help explain the classification of software requirements. A digital game called CLASSIFIQUI was developed using technologies such as *React Native* and *Firebase*. An evaluation was carried out with the students to collect data on the proposed game. The evaluation was carried out with students from the Federal University of Ceará - Campus Quixadá of the Software Engineering and Information Systems courses. The results obtained indicate that the CLASSIFIQUI game can be useful for the practical teaching of requirements classification.

**Keywords:** Requirements Engineering. Computer Games. Education.

## LISTA DE FIGURAS

Figura 1 – Processos de Verificação e Validação diálogo com NPC responsável no jogo.	22
Figura 2 – Tela durante a coleta de requisitos.	23
Figura 3 – Visão Geral dos Procedimentos Metodológicos.	30
Figura 4 – Aplicações desenvolvidas com o <i>React Native</i> .	32
Figura 5 – Cartas do Sistema Adote.	38
Figura 6 – Cartas do Sistema GREAT Print.	39
Figura 7 – Cartas do Sistema GREATOUR.	40
Figura 8 – Cartas do Sistema CAFETERIA.	41
Figura 9 – Cartas Tipos de Requisitos.	43
Figura 10 – Cartas Ajudas.	44
Figura 11 – Cartas Bônus.	44
Figura 12 – Tela do Ranking do Jogo.	45
Figura 13 – Gráfico do Conhecimento Prévio	47
Figura 14 – Gráfico da Experiência do Usuário - Parte 1	48
Figura 15 – Gráfico da Experiência do Usuário - Parte 2	49
Figura 16 – Tipos de Requisitos	58
Figura 17 – Telas do Aplicativo do Jogo.	60
Figura 18 – Tela Inicial do Jogo.	64
Figura 19 – Tela de Criar Conta no Jogo.	65
Figura 20 – Tela de Login do Jogo.	65
Figura 21 – Tela de Criar Sala no Jogo.	66
Figura 22 – Tela de Selecionar o Sistema no Jogo.	66
Figura 23 – Tela Esperando Amigos.	67
Figura 24 – Tela Inicial da Classificação.	67

## LISTA DE QUADROS

Quadro 1 – Definições dos Diferentes Tipos Requisitos de Software . . . . .	18
Quadro 2 – Comparativo entre os trabalhos relacionados e este trabalho. . . . .	28
Quadro 3 – Tipos de Requisitos Conhecidos dos Participantes . . . . .	48
Quadro 4 – Problemas Encontrados pelos Participantes . . . . .	50
Quadro 5 – Dificuldades ou Desvantagens Encontrados pelos Participantes . . . . .	50
Quadro 6 – Sugestões Dadas Pelos Participantes . . . . .	51

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
<b>1.1</b>	<b>Objetivos</b>	<b>15</b>
<b>1.1.1</b>	<b>Objetivo Geral</b>	<b>15</b>
<b>1.1.2</b>	<b>Objetivos Específicos</b>	<b>15</b>
<b>1.2</b>	<b>Organização</b>	<b>16</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>17</b>
<b>2.1</b>	<b>Requisitos de Software</b>	<b>17</b>
<b>2.2</b>	<b>Jogos</b>	<b>19</b>
<b>2.2.1</b>	<b>Jogos Sérios</b>	<b>20</b>
<b>2.2.1.1</b>	<b>Definição</b>	<b>21</b>
<b>2.2.1.2</b>	<b>Subgrupos</b>	<b>21</b>
<b>2.2.1.3</b>	<b>Características</b>	<b>21</b>
<b>2.2.1.4</b>	<b>Exemplos de Jogos Sérios</b>	<b>22</b>
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>24</b>
<b>3.1</b>	<b>Levantamento Bibliográfico</b>	<b>24</b>
<b>3.2</b>	<b>UbiRE: A Game for Teaching Requirements in the Context of Ubiquitous Systems (LIMA <i>et al.</i>, 2012)</b>	<b>24</b>
<b>3.3</b>	<b>Design and Preliminary Evaluation of a Cyber Security Requirements Education Game (SREG) (YASIN <i>et al.</i>, 2018)</b>	<b>25</b>
<b>3.4</b>	<b>Experiences of Using a Game for Improving Learning in Software Requirements Elicitation (GARCÍA <i>et al.</i>, 2019)</b>	<b>26</b>
<b>3.5</b>	<b>A serious Game for Teaching the Fundamentals of ISO/IEC/IEEE 29148 Systems and Software Engineering – Lifecycle Processes – Requirements Engineering at Undergraduate Level (GARCÍA <i>et al.</i>, 2020)</b>	<b>26</b>
<b>3.6</b>	<b>Avaliação da Aprendizagem em Experimentos com Jogo Educativo de Engenharia de Requisitos (GONÇALVES <i>et al.</i>, 2011)</b>	<b>27</b>
<b>3.7</b>	<b>Comparativo entre os Trabalhos Relacionados e este Trabalho</b>	<b>28</b>
<b>4</b>	<b>PROCEDIMENTOS METODOLÓGICOS</b>	<b>30</b>
<b>4.1</b>	<b>Criar o game design</b>	<b>30</b>
<b>4.2</b>	<b>Desenvolver o Jogo Digital</b>	<b>31</b>

4.3	<b>Avaliar o Jogo</b> . . . . .	32
5	<b>O JOGO: CLASSIFIQUI</b> . . . . .	34
5.1	<b>Público-Alvo</b> . . . . .	34
5.2	<b>Conhecimento Prévio</b> . . . . .	34
5.3	<b>Objetivos Educacionais</b> . . . . .	34
5.4	<b>Descrição do Jogo</b> . . . . .	34
5.5	<b>Gênero do Jogo</b> . . . . .	35
5.6	<b>Mecânica do Jogo</b> . . . . .	35
5.6.1	<i>Jogadores</i> . . . . .	35
5.6.2	<i>Cartas</i> . . . . .	35
5.6.3	<i>Regras</i> . . . . .	35
5.7	<b>Plataforma do Jogo</b> . . . . .	36
5.8	<b>Narrativa do Jogo</b> . . . . .	37
5.9	<b>Formas de Interação com o Jogo</b> . . . . .	37
5.9.1	<i>Deck de Desafios</i> . . . . .	37
5.9.1.1	<i>Adote</i> . . . . .	37
5.9.1.2	<i>GREAT Print</i> . . . . .	39
5.9.1.3	<i>GREATOUR</i> . . . . .	39
5.9.1.4	<i>Cafeteria</i> . . . . .	41
5.9.2	<i>Deck de Jogo</i> . . . . .	43
5.9.3	<i>Deck de Ajuda</i> . . . . .	43
5.9.4	<i>Deck de Bônus</i> . . . . .	44
5.10	<b>Sistema de Pontuação</b> . . . . .	45
6	<b>AVALIAÇÃO DO JOGO</b> . . . . .	46
6.1	<b>Planejamento da Avaliação</b> . . . . .	46
6.2	<b>Execução da Avaliação</b> . . . . .	46
6.3	<b>Resultados da Avaliação</b> . . . . .	47
7	<b>CONCLUSÃO</b> . . . . .	52
	<b>REFERÊNCIAS</b> . . . . .	53
	<b>APÊNDICE A–MAPEAMENTO DOS TIPOS DE REQUISITOS</b> . . . . .	56
	<b>APÊNDICE B–TELAS DO APLICATIVO DO JOGO</b> . . . . .	60
	<b>APÊNDICE C–QUESTIONÁRIO DE AVALIAÇÃO DO JOGO</b> . . . . .	62

**APÊNDICE D-UTILIZAÇÃO DO JOGO . . . . . 64**

## 1 INTRODUÇÃO

No mundo moderno em que vivemos, a utilização de produtos de software tornou-se cada vez mais necessária. Diversas áreas da sociedade estão utilizando softwares para gerenciar seus serviços. Sendo assim, a Engenharia de Software (ES) é essencial para o funcionamento da sociedade (SOMMERVILLE, 2011). Segundo Pressman e Maxim (2011), a ES abrange processos, métodos e uma variedade de ferramentas que possibilitam os profissionais desenvolverem software dentro do prazo e com qualidade.

Uma das áreas cruciais para o desenvolvimento de software é Engenharia de Requisitos (ER) (SOMMERVILLE, 2011), pois muitos dos problemas encontrados na entrega de um software são originados nessa área (WIEGERS; BEATTY, 2013).

Os requisitos de software são descrições do que o sistema deverá realizar (SOMMERVILLE, 2011; WIEGERS; BEATTY, 2013). A ER compreende todo o processo de descoberta, análise e validação desses requisitos. Em linhas gerais, a ER é subdividida em duas áreas: **desenvolvimento de requisitos**, onde destaca-se as etapas de elicitação, análise, especificação e validação; e o **gerenciamento de requisitos**, que engloba as etapas de manutenção da documentação e a gerência dos requisitos (SOMMERVILLE, 2011; ISO/IEC, 2018). O principal resultado obtido por meio da ER é o conjunto de requisitos documentados e validados (ISO/IEC, 2018).

Dessa forma, é de fundamental importância prezar pela qualidade do ensino da ER nos cursos de graduação na área da computação, tendo em vista que um conhecimento defasado sobre engenharia de requisitos poderá ocasionar problemas na entrega do software (NAVARRO-ALMANZA *et al.*, 2018). Além disso, o MEC ([S. l.], 2016) determina que os cursos de graduação na área de computação devem fornecer conhecimento necessário para os estudantes sobre requisitos.

O ensino da Engenharia de Software, e, assim, da Engenharia de Requisitos, está evoluindo e novas abordagens estão surgindo (SILVA J., 2018). Uma que está ganhando espaço e visibilidade é a utilização de jogos para os alunos colocarem em prática o que aprenderam de forma teórica, dado que a aprendizagem por meio de jogos é mais interessante e divertida para os estudantes (SAVI *et al.*, 2011).

De acordo com Sakuda e Fortim (Brasília: Ministério da Cultura, 2018), os jogos digitais ganharam destaque importante em diversas áreas, tais como educação, treinamento e simulação de situações profissionais. Sendo assim, o desenvolvimento de um jogo sério

baseado no formato de digital torna-se uma abordagem interessante para o ensino de requisitos de software, em especial para a fase de Análise de Requisitos, que é onde a classificação das informações obtidas na elicitação em tipos de requisito é realizada. Classificar as informações que foram adquiridas do cliente é o início do processo para criar especificações de requisitos (SOMMERVILLE, 2011; WIEGERS; BEATTY, 2013) e é o foco desse trabalho.

No que diz respeito ao uso de jogos sérios para o ensino de requisitos, foi realizada uma busca para encontrar trabalhos que se relacionassem com a análise de requisitos em bases de dados e uma das bases utilizadas foi a *Scopus*<sup>1</sup>. Estudos que propõem jogos para resolverem diferentes problemas da engenharia de requisitos foram identificados.

Nessa busca, não foram encontrados trabalhos com foco específico na fase *Análise de Requisitos*, na qual os analistas irão refinar os requisitos de software e classificar as informações obtidas da elicitação em diversos tipos de requisitos, como por exemplo, requisito de usuário, requisito funcional, requisito não-funcional, entre outros (WIEGERS; BEATTY, 2013).

Assim, isso reforça a necessidade de apoiar o ensino da Engenharia de Requisitos na fase de Análise de Requisitos através da construção de um jogo. Tendo em vista que um ensino defasado sobre este assunto poderá ocasionar problemas na produção de software.

## **1.1 Objetivos**

Nesta Seção, os objetivos desse trabalho são apresentados.

### **1.1.1 Objetivo Geral**

O objetivo geral deste trabalho é desenvolver um jogo sério digital para apoiar o ensino da classificação de requisitos em cursos de graduação da área de computação.

### **1.1.2 Objetivos Específicos**

O objetivo geral desse trabalho foi refinado em três objetivos específicos listados a seguir.

- a. Criar o game design
- b. Implementar o jogo na plataforma *Android* seguindo o game design estabelecido no objetivo específico anterior

---

<sup>1</sup> <https://www.scopus.com/home.uri>, acesso em: 19 de Abril de 2020

c. Avaliar o jogo com alunos do Campus de Quixadá

## **1.2 Organização**

Esse trabalho está organizado em mais seis capítulos, descritos a seguir. O Capítulo 2 apresenta os conceitos que fundamentam o desenvolvimento do presente trabalho. No Capítulo 3 são apresentados cinco trabalhos relacionados com o presente trabalho. O Capítulo 4 exhibe a metodologia utilizada nesse trabalho. No Capítulo 5, o jogo proposto chamado CLASSIFIQUI é apresentado. No Capítulo 6, a avaliação do jogo é apresentada. Por fim, no Capítulo 7, a conclusão e os trabalhos futuros são apresentados e discutidos.

## 2 REFERENCIAL TEÓRICO

Nesta Seção são apresentados os conceitos que fundamentam o desenvolvimento desse trabalho. Na Seção 2.1, são introduzidos os conceitos sobre Requisitos de Software, descrevendo suas definições e suas diferentes classificações. Na Seção 2.2, são apresentados conceitos sobre jogos, desde a sua definição até categorias de jogos digitais, também são apresentados assuntos relacionados a jogos sérios, tais como elementos de designer para a construção de um jogo sério, categorias e exemplos.

### 2.1 Requisitos de Software

Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada. Por exemplo, em um sistema de controle de vendas, é natural que os clientes necessitem controlar um dispositivo, colocar um pedido ou encontrar informações (SOMMERVILLE, 2011).

Sommerville (2011), classifica os requisitos de software em duas categorias principais sendo elas requisitos funcionais e requisitos não-funcionais. Ainda de acordo com o autor, requisitos funcionais (RFs) são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações. Já os requisitos não-funcionais (RNFs) são restrições aos serviços ou funções oferecidos pelo sistema. Incluem restrições de *timing*, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não-funcionais, muitas vezes, aplicam-se ao sistema como um todo.

Os requisitos de software podem ser classificados em RFs, requisitos de usuário, requisitos de sistema, regras de negócio, entre outros. Sommerville (2011) classifica os requisitos não-funcionais em três sub-categorias, são elas: (i) requisitos de produto, os quais especificam ou restringem o comportamento do software; (ii) requisitos de organização, onde estão os requisitos gerais de sistemas derivados das políticas e procedimentos da organização do cliente e do desenvolvedor e (iii) requisitos externos que abrangem todos os requisitos que derivam de fatores externos ao sistema e seu processo de desenvolvimento. Podem incluir requisitos reguladores, que definem o que deve ser feito para que o sistema seja aprovado para uso, por um

regulador, tal como um banco central; requisitos legais que devem ser seguidos para garantir que o sistema opere dentro da lei e requisitos éticos que asseguram que o sistema será aceitável para seus usuários e o público em geral.

Existem diferentes tipos de informações a respeito de requisitos de software, tanto que é necessário um conjunto de adjetivos, ou seja, diferentes termos para a classificação dos requisitos de software, para não deixar sobrecarregado o termo requisito (WIEGERS; BEATTY, 2013). Sendo assim a classificação proposta pelo Sommerville seria ineficiente para o desenvolvimento do trabalho, por este motivo será utilizada a classificação apresentada no Quadro 1, que foi proposta por Wieggers e Beatty (2013).

Quadro 1 – Definições dos Diferentes Tipos Requisitos de Software

Tipo	Definição
Requisitos de Negócio	Um objetivo comercial de alto nível da organização que cria um produto ou de um cliente que o adquire.
Regras de Negócio	Uma política, diretriz, padrão ou regulamento que define ou restringe algum aspecto do negócio. Não é um requisito de software em si, mas a origem de vários tipos de requisitos de software.
Restrições	Uma restrição imposta às opções disponíveis para o desenvolvedor para o design e construção de um produto
Requisitos de Interface Externa	Uma descrição de uma conexão entre um sistema de software e um usuário, outro sistema de software ou um dispositivo de hardware.
<i>Feature</i>	Um ou mais recursos do sistema relacionados à lógica que agregam valor ao usuário e são descritos por um conjunto de requisitos funcionais.
Requisitos Funcionais	Uma descrição de um comportamento que um sistema exibirá sob condições específicas.
Requisitos Não-Funcionais	Uma descrição de uma propriedade ou característica que um sistema deve exibir ou uma restrição que ele deve respeitar.
Atributos de Qualidade	Um tipo de requisito não-funcional que descreve um serviço ou característica de desempenho de um produto.
Requisitos de Sistema	Um requisito de nível superior para um produto que contém vários subsistemas, que podem ser todos software ou software e hardware.
Requisitos de Usuário	Uma meta ou tarefa que classes específicas de usuários devem poder executar com um sistema ou com um atributo de produto desejado.

Fonte: Adaptado de (WIEGERS; BEATTY, 2013).

Dado as definições citadas acima, a classificação de requisitos pode parecer uma

tarefa simples da engenharia de requisitos, porém não é tão simples assim pois alguns requisitos elicitados pelo analista poderá ser classificado em mais de uma categoria e assim poderá vir a ocasionar dúvidas ao analista em qual categoria o mesmo deverá ser classificado. Em muitos projetos é necessário priorizar quais requisitos serão desenvolvidos primeiro, sendo assim é necessário saber a classificação do requisito para definir a sua prioridade, pois algumas regras de negócio devem vir a serem classificadas de forma correta, pois na maioria dos casos a não classificação correta delas poderá vir a causar transtornos no desenvolvimento do projeto (WIEGERS; BEATTY, 2013).

## 2.2 Jogos

Para Michaelis (2016), jogos representam qualquer atividade recreativa que tem por finalidade entreter, divertir ou distrair. Os jogos possuem diferentes características que os distinguem uns dos outros, por exemplo, os jogos sérios têm como característica principal ensinar algo de forma lúdica, divertida e não somente entreter os usuários (AYDAN *et al.*, 2017)

Contudo as características diferentes que os jogos possuem, torna necessário uma classificação para os mesmos. Sendo assim, muitos autores definem categorias para jogos (SILVA A., 2010; BRITO, 2016).

Para Prensky (2004), os jogos podem ser classificados de acordo com seu gênero, da seguinte forma:

1. **Jogos de Ação:** são caracterizados pelo tempo de reação e a coordenação olhos-mãos do jogador, jogos de ação são experiências em tempo real e com ênfase na restrição de tempo das respostas do jogador em relação as tarefas físicas.
2. **Jogos de Estratégia:** caracterizam-se por serem focados em táticas e planejamento, como também no gerenciamento de unidades e recursos. Podem ser divididos em estratégia por turnos e estratégia em tempo real. Os jogos de estratégia contemplam temas como conquista, exploração e comércio.
3. **Jogos de Esportes:** são simulações simplificadas dos esportes. Muitos desses jogos envolvem times, temporadas e torneios, além de muitos também se basearem em regras, características estéticas e títulos dos esportes do mundo real.
4. **Jogos de Corrida/Condução:** são simulações do mundo real ou de um mundo fantasiado, onde ambos têm em comum o fato de um personagem está correndo sob controle do jogador. Esse jogo deve dispor as câmeras de um modo que forneça a ilusão da velocidade

e do controle.

5. **Jogos de Voo e outros Simuladores:** são simulações como jogos de ação, porém baseados em atividades do mundo real. Esses simuladores são complexos, pois buscam aproximar o jogador de uma experiência real, com instrumentação e controles complexos.
6. **Jogos de Aventura:** são jogos com ênfase na exploração de um mundo, coleta de itens e solução de quebra-cabeças, podendo ser combinado com elementos de ação. Concentram-se em personagens, porém sem possibilidade de evolução e sem acúmulo de experiência.
7. **RPG, do inglês *Role-Playing Games*:** caracteriza-se pela criação e evolução de personagens, essa evolução muitas vezes ocorre ao mesmo tempo que o controle de seu inventário, a exploração de mundos e o acúmulo de riquezas, status e experiência. Esses gêneros de jogos iniciam e terminam com o mesmo personagem e tendem a ter missões para esse personagem, essas missões são vinculadas a história que é contada durante o jogo.
8. **Jogos de Simulação/Construção:** são pequenas representações de ambientes reais, sistemas do mundo real, possibilitando que o jogador gerencie seu negócio virtual, são jogos com foco no gerenciamento e na manutenção de certos recursos, juntamente com a construção de algo.
9. **Jogos *Multiplayer*:** são jogos que permitem que vários jogadores participem simultaneamente de uma mesma partida, interagindo entre si.

Além desses tipos de jogos apresentados anteriormente, temos também os jogos sérios que além de apresentar as características dos jogos anteriormente citados, apresenta também uma característica específica que é a de ensinar os jogadores através do jogo (AYDAN *et al.*, 2017).

### **2.2.1 Jogos Sérios**

Nesta Seção a teoria sobre jogos sério é apresentada. Na Subseção 2.2.1.1 temos conceitos sobre o que são jogos sérios. Na Subseção 2.2.1.2, são apresentados subgrupos nos quais os jogos sérios podem ser divididos. Na Subseção 2.2.1.3 são apresentadas características dos jogos sérios. Na Subseção 2.2.1.4, são apresentados dois exemplos de jogos sérios desenvolvidos para a educação de diferentes áreas do conhecimento.

### 2.2.1.1 Definição

Segundo Aydan *et al.* (2017), jogo sério é uma abordagem interativa projetada para um propósito diferente de puro entretenimento. O objetivo de um jogo sério é geralmente melhorar um aspecto educacional onde os participantes certamente frequentam tais atividades com tal expectativa. Para Michael e Chen (2006) um jogo sério é um jogo no qual a educação (em suas várias formas) é o objetivo principal, e não o entretenimento.

Atualmente os jogos sérios estão sendo utilizados como meios de ensino em diferentes áreas da educação, dentre elas podemos destacar a medicina, engenharia de software, engenharia civil, entre outras. Para (SAVI; ULBRICHT, 2008), os jogos digitais já são utilizados em diferentes níveis de ensino, indo desde a pré escola até cursos de graduação, especializações e cursos corporativos.

### 2.2.1.2 Subgrupos

Alguns autores chegaram a classificar os jogos sérios em subgrupos distintos pois para cada subgrupo o mesmo foca em alguma área de estudo, abaixo são listados três subgrupos de jogos sérios definidos por (ROCHA; ARAUJO, 2013)

- Simulações de treinamento refletem o comportamento de um sistema, mas não incluem necessariamente gráficos realistas (diferente de jogos);
- Jogos educacionais são jogos simples que não simulam sistemas (diferente de simulações);
- e
- Jogos de simulações são jogos que possibilitam experiências reais mas sem os aspectos de aprendizagem ou treinamento.

### 2.2.1.3 Características

Segundo Soares (2012), os jogos sérios possuem a capacidade e a característica de trazer aos seus usuários diversos benefícios na sua utilização no que se refere ao processo de desenvolvimento educacional.

Atividades relacionadas a um determinado assunto são desenvolvidas e ao finalizar elas, questionários ou entrevistas são realizados para averiguar se o conteúdo apresentado pelo jogo foi o suficiente para o jogador aprender (AYDAN *et al.*, 2017), ou seja, para verificar se atingiu o objetivo de beneficiar o processo educacional.

### 2.2.1.4 Exemplos de Jogos Sérios

Com o passar do tempo e o avanço da tecnologia viu-se a necessidade de se adaptar a esse novo mundo, sendo assim muitos pesquisadores começaram a ver nos jogos uma oportunidade para auxiliar no desenvolvimento na aprendizagem de seus alunos por esse e outros motivos surgiram diversos jogos sérios para diferentes áreas do conhecimento, abaixo são citados quatro exemplos de jogos sérios: (HAINEY *et al.*, 2011; BRITO, 2016; COLPANI; FARIA, 2017; AYDAN *et al.*, 2017; BEPPE *et al.*, 2018)

1. Floors - Jogo desenvolvido para auxiliar no ensino da norma ISO/IEC 12207 (AYDAN *et al.*, 2017). A Figura 1, apresenta a tela desse jogo e se refere ao diálogo a respeito dos processos de verificação e validação.

Figura 1 – Processos de Verificação e Validação diálogo com NPC responsável no jogo.



Fonte: Adaptado de Aydan *et al.* (2017)

O jogo foi desenvolvido com intuito de auxiliar estudantes no aprendizado da ISO/IEC 12207. No jogo é criado um ambiente de escritório virtual e uma organização onde a gestão de projetos de software e o processo ISO/IEC 12207 podem ser realizados.

2. Jogo desenvolvido para auxiliar no ensino da coleta e análise de requisitos de software (HAINEY *et al.*, 2011). A Figura 2, apresenta a tela desse jogo, onde os personagens presentes no jogo interagem entre si para coletar requisitos.

Figura 2 – Tela durante a coleta de requisitos.



Fonte: Adaptado de Hainey *et al.* (2011)

O jogo foi desenvolvido para auxiliar na aprendizagem da coleta e análise de requisitos de software. A ideia básica do jogo é que a equipe gere e entregue os projetos de software. Cada jogador tem uma função específica, como gerente de projeto, analista de sistemas, designer de sistemas ou líder de equipe.

O presente trabalho abordará os diferentes tipos de requisitos de software, no que diz respeito à classificação dos mesmos através de um jogo sério capaz de auxiliar o aluno nessa tarefa, o jogo será desenvolvido utilizando o *react native* para a plataforma android.

### 3 TRABALHOS RELACIONADOS

Neste Capítulo serão apresentados os trabalhos relacionados identificados e como foi realizada essa busca.

#### 3.1 Levantamento Bibliográfico

A busca por trabalhos relacionados foi feita na base de dados *Scopus*<sup>1</sup>, através da *string* de busca: *(TITLE-ABS-KEY ( "Software Requirements"OR "Requirements Engineering") AND TITLE-ABS-KEY ( teaching OR education ) AND TITLE-ABS-KEY ( game ) ) AND ( LIMIT-TO ( SUBJAREA , "COMP" ) ) AND ( LIMIT-TO ( LANGUAGE , "English" ) OR LIMIT-TO ( LANGUAGE , "Portuguese" ) )*.

Utilizando a *string* diretamente na base, cinquenta e três trabalhos foram retornados. Dentre entre esses trabalhos, foram selecionados os que continham as palavras *Game* e *Software Requirements* ou *Requirements Engineering* em seu título ou resumo, sendo assim ficaram treze trabalhos, desses depois de realizada uma leitura da introdução e dos resultados obtidos pelos trabalhos, foram selecionados os trabalhos apresentados a seguir.

#### 3.2 UbiRE: A Game for Teaching Requirements in the Context of Ubiquitous Systems (LIMA *et al.*, 2012)

UbiRE (LIMA *et al.*, 2012) foi um jogo desenvolvido como forma de ensinar a Engenharia de Requisitos (ER) no contexto de sistemas ubíquos de forma divertida. O jogo acontece em um ambiente virtual 3D representado por uma casa, onde o mesmo está dividido em 4 (quatro) fases, sendo estas representadas pelos cômodos da casa. O jogador é responsável por montar o sistema da casa, onde ele deve criar as conexões ubíquas entre os equipamentos presentes em cada cômodo além do que essas conexões representam os níveis de dificuldade do jogo, pois a cada nível existe um número mínimo de conexões necessárias para passar para o próximo.

Com o intuito de verificar a validade do jogo, inicialmente os autores realizaram uma avaliação de usabilidade, onde os participantes descreviam os pontos fortes e fracos do jogo a fim de auxiliar na conclusão do desenvolvimento do jogo.

Após a realização de experimentos com estudantes utilizando o jogo, os autores

<sup>1</sup> <https://www.scopus.com/home.uri>, acesso em: 19 de Abril de 2020

concluíram que a utilização do jogo para o ensino da ER no contexto de sistemas ubíquos foi satisfatório. No entanto, os autores fizeram um estudo apenas relacionando a engenharia de requisitos no contexto de sistemas ubíquos, não levaram em questão a engenharia de requisitos em um contexto geral.

Sendo assim o trabalho de Lima *et al.* (2012) se relaciona com esse trabalho de TCC no sentido de ensinar engenharia de requisitos. A diferença é que o trabalho de TCC irá tratar de uma fase específica da ER, a classificação de requisitos. E essa será feita por meio de um jogo digital de cartas *multiplayer*.

### **3.3 Design and Preliminary Evaluation of a Cyber Security Requirements Education Game (SREG) (YASIN *et al.*, 2018)**

SREG (YASIN *et al.*, 2018) é um jogo de cartas *multiplayer*, desenvolvido para instruir a respeito de requisitos de segurança. O jogo se concentra na cooperação dos jogadores dentro dele, para conseguir melhores resultados a respeito da aprendizagem sobre requisitos de segurança.

Ao iniciar o jogo todos os jogadores recebem o papel de atacante, pois o jogo se baseia em ataques a organizações, sejam eles ataques internos ou externos à segurança dos sistemas da organização. Os atacantes devem definir o alvo, ou seja, a organização que será atacada e definem o valor agregado (VA) do ataque. Para definir o VA, é necessário que algum dos atacantes da equipe tenha conhecimento sobre o assunto.

Por fim, os autores realizaram uma avaliação empírica do jogo para comprovar a experiência satisfatória dos participantes. Segundo Yasin *et al.* (2018), a solução baseada em jogos pode ser um caminho alternativo para as metodologias de ensino.

O trabalho proposto por Yasin *et al.* (2018), apresenta características muito semelhantes a proposta por este trabalho, pois o jogo desenvolvido pelos autores é um jogo de cartas e *multiplayer*, algo que será utilizado neste trabalho. Porém o jogo trabalha apenas com requisitos de segurança e este trabalho tem por objetivo trabalhar requisitos de software no geral, principalmente no que diz respeito à classificação de requisitos de software em diversos tipos, além do mais o trabalho que será desenvolvido não irá se tratar de um jogo não digital de cartas diferentemente do que foi proposto por Yasin *et al.* (2018) em seu trabalho.

### **3.4 Experiences of Using a Game for Improving Learning in Software Requirements Elicitation (GARCÍA *et al.*, 2019)**

García *et al.* (2019) desenvolveram o jogo *Biyubi* com o intuito de auxiliar no ensino das técnicas de elicitação de requisitos de software nos cursos de graduação em engenharia. O jogo funciona como uma simulação do mundo real onde os alunos interagem com outros personagens do jogo, para aprenderem as técnicas de elicitação sem a presença do professor.

O jogo se passa dentro de um ambiente de uma biblioteca, onde deverão ser elicitados requisitos para construir o sistema da mesma, para isso são realizadas atividades de elicitação de requisitos. Durante a realização dessas atividades, os jogadores assumem papéis com uma função predeterminada. Um exemplo são os Alunos principais usuários da biblioteca que podem fornecer uma quantidade considerável de informações sobre isso. Alguns alunos podem compartilhar informações após um breve desafio, que deverá ser completado pelo jogador.

Para validar o jogo foram feitos estudos de caso nas turmas do curso de graduação em engenharia de software, um deles foi o curso de engenharia de software, onde o principal objetivo era avaliar a satisfação, a motivação e a experiência de aprendizagem dos alunos. Os autores obtiveram como resultado que os alunos apresentaram uma melhoria significativa na aplicação da teoria na prática do jogo.

García *et al.* (2019) trabalha apenas com as técnicas utilizadas para a elicitação de requisitos, diferentemente do trabalho que será desenvolvido no TCC, que busca auxiliar no ensino da classificação de requisitos.

### **3.5 A serious Game for Teaching the Fundamentals of ISO/IEC/IEEE 29148 Systems and Software Engineering – Lifecycle Processes – Requirements Engineering at Undergraduate Level (GARCÍA *et al.*, 2020)**

García *et al.* (2020) desenvolveram o jogo *Requengin*. Nele são implementadas as técnicas referentes a engenharia de requisitos propostos pela ISO/IEC 29148. Assim, ele foi desenvolvido com a proposta de melhorar a compreensão a respeito da norma.

O jogo funciona como uma simulação em 3D onde os estudantes são ensinados sobre os fundamentos da engenharia de requisitos do ponto de vista da norma ISO/IEC 29148. Usando a perspectiva de exploração do mundo real, os alunos precisam explorar três cenários simulados, são eles a casa do jogador, a sala de aula da universidade e a biblioteca acadêmica, para aprender

a executar os processos de engenharia de requisitos.

O jogo trabalha com duas principais atividades da engenharia de requisitos definidas pela ISO, que são a elicitação de requisitos com os *stakeholders* e a análise de requisitos. Para a atividade de elicitação, o jogo fornece 53 personagens do jogo não jogáveis, ou seja, o jogo é quem controla eles e não jogadores externos ao jogo, com quem os alunos devem interagir ativamente para realizar a elicitação.

Para validar o jogo, foi feita uma avaliação empírica com os participantes. Com a avaliação, foi constatado que as atividades propostas pelo jogo ajudaram a fortalecer a compreensão da teoria dos conceitos aprendidos em sala de aula. O jogo trabalha com algumas atividades da engenharia de requisitos, porém nenhuma delas envolve a classificação de requisitos de software, que é o que este trabalho se propõe a desenvolver.

### **3.6 Avaliação da Aprendizagem em Experimentos com Jogo Educativo de Engenharia de Requisitos (GONÇALVES *et al.*, 2011)**

Gonçalves *et al.* (2011) desenvolveram o jogo Ilha dos Requisitos. O jogo foi criado com o intuito de apoiar o ensino da Engenharia de Requisitos no contexto geral. Além disso, o jogo tem como finalidade transmitir ao jogador a importância de aplicar adequadamente a Engenharia de Requisitos em projetos de software.

O jogo funciona em ambiente representado por uma ilha onde o jogador é representado por um personagem o Jack Reqs, e este enfrenta diversos desafios na ilha. Sendo o objetivo do jogo ajudar Jack a sair da ilha antes que o vulcão existente na ilha entre em erupção e o mate.

Os pontos dos desafios são marcados no mapa da ilha porém só são habilitados quando Jack já realizou os desafios anteriores que possibilitam o desbloqueio do próximo. O Jack possui um notebook onde ele pode solicitar dicas para a resolução dos desafios.

Gonçalves *et al.* (2011) realizaram validações empíricas para comprovar a validade do jogo quanto ao ensino da Engenharia de Requisitos. Onde foram realizados dois experimentos, onde existia uma preocupação quanto aos dados coletados e sua precisão, sendo assim foi utilizado um teste não paramétrico (Mann-Whitney unilateral U) para avaliar as hipóteses sobre a efetividade de aprendizagem do jogo. Os resultados qualitativos apontam que os alunos ficaram motivados a resolver os desafios do jogo e além disso se propuseram a utilizar o jogo novamente.

O jogo desenvolvido pelos autores Gonçalves *et al.* (2011) parece bastante com o presente trabalho pois o mesmo apresenta desafios, ajudas para o jogador, além de ao final

de cada partida apresenta um ranking com a pontuação dos jogadores, algo que também foi implementado no presente trabalho.

### 3.7 Comparativo entre os Trabalhos Relacionados e este Trabalho

Nesta seção é apresentado o Quadro 2, na qual um comparativo entre os trabalhos relacionados com este trabalho é apresentado. A comparação é feita através dos seguintes critérios: (i) Propõe um jogo de cartas: foi analisados nos trabalhos se os mesmos desenvolveram jogos baseados em jogos de cartas, tendo em vista que o presente trabalho será desenvolvido um jogo de cartas; (ii) Propõe um jogo digital: foi analisados nos trabalhos se os mesmo desenvolveram jogos digitais, tendo em vista que o presente trabalho será desenvolvido um jogo digital; (iii) Propõe um jogo onde os jogadores trabalham em equipe: os jogos sérios possuem como características a interatividade entre os jogadores, sendo assim é de importância que o jogo desenvolvido os jogadores possam trabalhar em equipe; (iv) Realizou(ará) mapeamento dos tipos de requisitos existentes na literatura: pelo presente trabalho se propor a desenvolver um jogo sobre classificação de requisitos torna necessário o mapeamento dos tipos de requisitos e (v) Realizou(ará) experimentos para a validação do trabalho: para validar o jogo é necessário realizar experimentos com o público alvo, pois os mesmos devem ser os principais interessados no desenvolvimento do jogo.

Quadro 2 – Comparativo entre os trabalhos relacionados e este trabalho.

	Lima <i>et al.</i> (2012)	Yasin <i>et al.</i> (2018)	García <i>et al.</i> (2019)	García <i>et al.</i> (2020)	Gonçalves <i>et al.</i> (2011)	Presente trabalho
Propõe um jogo de cartas	Não Atende	Atende	Não Atende	Não Atende	Não Atende	Atende
Propõe um jogo digital	Atende	Não Atende	Atende	Atende	Atende	Atende
Propõe um jogo onde os jogadores interagem entre si	Não Atende	Atende	Não Atende	Não Atende	Atende	Atende
Realizou(ará) mapeamento dos tipos de requisitos existentes na literatura	Não Atende	Não Atende	Não Atende	Não Atende	Não Atende	Atende
Realizou(ará) experimentos para a validação do trabalho	Atende	Atende	Atende	Atende	Atende	Atende

Fonte: Elaborado pelo Autor.

Por meio do Quadro 2, fica clara as diferenças entre os trabalhos apresentados com o presente trabalho no que diz respeito a interatividade entre os jogadores. Este trabalho propõe um jogo onde será possível os jogadores discutir entre si sobre classificação de requisitos presentes no mesmo e assim acontecer uma interatividade maior entre os jogadores, tendo em vista que entre os trabalhos apenas os autores Yasin *et al.* (2018) propõem um jogo em equipe ou que os jogadores estivessem presentes em um mesmo ambiente.

Além disso, observando o Quadro 2 percebemos que apesar de os trabalhos propostos

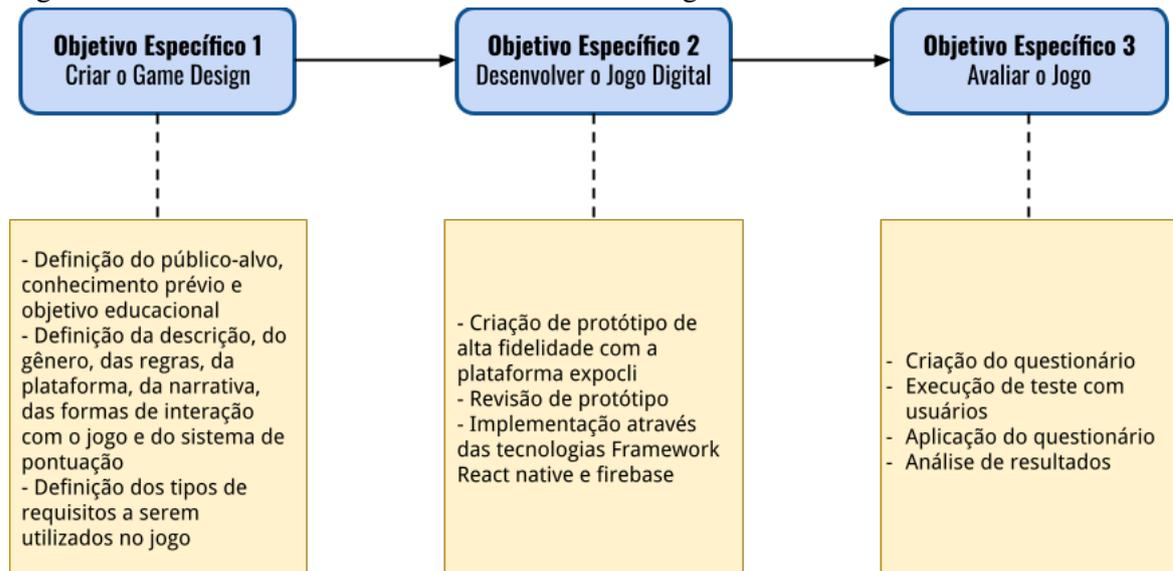
por Lima *et al.* (2012), García *et al.* (2019) e García *et al.* (2020) se tratarem de jogos digitais os mesmos não são jogos de cartas, o que o presente trabalho busca desenvolver.

O presente trabalho se propõe a realizar um mapeamento dos tipos de requisitos presentes na literatura para definir o conteúdo do jogo que irá ser desenvolvido, sendo que os demais trabalhos não realizaram este mapeamento pois seus jogos apresentam conteúdo de tipos específicos de requisitos.

## 4 PROCEDIMENTOS METODOLÓGICOS

Este trabalho está organizado em três grandes fases, seguindo os três objetivos específicos. Cada um destes objetivos é apoiado através de passos metodológicos. A Figura 3 apresenta uma visão geral dos objetivos e seus respectivos passos.

Figura 3 – Visão Geral dos Procedimentos Metodológicos.



Fonte: Elaborado pelo Autor

A seguir esses passos são detalhados.

### 4.1 Criar o game design

O game design foi desenvolvido baseado nas definições sobre jogos sérios proposto por SILVA A. (2010). Segundo esse mesmo autor, para considerar um jogo sério como educacional devemos definir três pontos no game design, são eles (i) Público alvo; (ii) Conhecimentos prévios necessários pelos jogadores para realizarem as atividades propostas pelo jogo; e (iii) Objetivos educacionais nos quais referem-se ao conteúdo proposto pelo jogo e qual a sua finalidade de ensino.

Seguindo a proposta de SILVA A. (2010), o primeiro passo metodológico para a criação do game design foi a (i) Definição do público-alvo, conhecimento prévio e objetivo educacional.

Para Brathwaite e Schreiber (2009) game design funciona como o processo de criação do conteúdo e das regras de um jogo, onde um bom game design consiste em gerar objetivos que

motivem o jogador a alcançá-los e às regras, que este deve seguir enquanto realiza decisões para alcançar os objetivos.

Sendo assim, o segundo passo metodológico para a criação do game design foi (ii) Definição da descrição, do gênero, das regras, da plataforma, da narrativa, das formas de interação com o jogo e do sistema de pontuação.

Como SILVA A. (2010) fala em seu trabalho que um jogo deve possuir um objetivo educacional para que seja considerado educativo, sentiu-se a necessidade de realizar a (iii) Definição dos tipos de requisitos a serem utilizados no jogo, tornando-se assim o terceiro e último passo metodológico para a criação do game design.

## 4.2 Desenvolver o Jogo Digital

Para realizar o desenvolvimento do jogo foi seguido alguns passos metodológicos, são eles: (i) Criação de protótipo de alta fidelidade com a plataforma expocli, (ii) Revisão de protótipo e (iii) Implementação através das tecnologias Framework React native e firebase.

Inicialmente foi construído um protótipo de alta fidelidade do jogo utilizando a plataforma expocli com as principais funcionalidades descritas no game design. Após a criação do protótipo, foi realizada uma coleta de opinião com dois usuários. Para Prates e Barbosa (2003), a coleta de opinião dos usuários busca identificar a apreciação dos usuários a respeito do sistema desenvolvido, no caso do presente trabalho o jogo desenvolvido. Foram coletadas desses usuários opiniões a respeito do designer do jogo, exemplo a cor das cartas poderia ser melhorada, as cartas de requisitos poderiam ser embaralhadas para melhorar a dinâmica do jogo.

Tendo em vista a coleta, neste processo foram identificados dois aspectos a respeito do jogo, foram eles, (i) se a aparência estética do jogo é satisfatória e (ii) se eles gostariam de utilizar o jogo quando estiver desenvolvido completamente.

Por fim, depois de coletados os dados a respeito destes dois aspectos sobre as cartas do jogo através de *feedback* dos usuários, os mesmos foram analisados e de acordo com as principais sugestões de modificação relatadas pelos participantes da coleta foi desenvolvido o jogo utilizando o framework *react native* e o *firebase*.

*React*<sup>1</sup> é um *framework* desenvolvido para melhorar o desenvolvimento de interfaces de usuário das aplicações. Os componentes que estão presentes no *react* foram desenvolvidos pelo Facebook.

<sup>1</sup> <https://pt-br.reactjs.org/>, acesso em: 10 de Fevereiro de 2021

O *React Native*<sup>2</sup> é um *framework* para desenvolvimento de aplicativos móveis multiplataforma. O *react native* é baseado no *react*, que é uma ferramenta para desenvolvimento de aplicações *javascript* para a web, e foi desenvolvido para facilitar a criação de aplicações multiplataforma mobile utilizando apenas *javascript*.

Segundo dados do site oficial do *react native*<sup>1</sup>, em 2018 este tinha o segundo maior número de colaboradores de repositórios quaisquer do Github<sup>3</sup>, empresas como Callstack, Expo, Infinite Red, Microsoft e Software Mansion, são algumas das principais colaboradoras do *react native*. A Figura 4 apresenta algumas aplicações desenvolvidas com *react native*

Figura 4 – Aplicações desenvolvidas com o *React Native*.



Fonte: Adaptado de *React Native*<sup>1</sup>

*Firebase*<sup>1</sup> é um ferramenta desenvolvida e mantida pelo Google, ela foi desenvolvida com o intuito de melhorar o desenvolvimento de aplicações web e mobile.

### 4.3 Avaliar o Jogo

Para realizar a avaliação do jogo foram seguidos alguns passos metodológicos, são eles: (i) Criação do questionário, (ii) Execução de teste com usuários, (iii) Aplicação do questionário e (iv) Análise de resultados.

Esta etapa foi realizada a avaliação do jogo por meio de uma avaliação qualitativa. Para Prates e Barbosa (2003), dados qualitativos são resultados não numéricos obtidos, tais como uma lista de problemas que os usuários tiveram ao utilizar a aplicação, ou suas sugestões sobre como melhorar o sistema, neste caso o jogo.

<sup>2</sup> <https://reactnative.dev/>, acesso em: 10 de Fevereiro de 2021

<sup>3</sup> <https://github.com/>, acesso em 11 de Fevereiro de 2021

<sup>1</sup> <https://firebase.google.com/docs>, acesso em: 10 de Fevereiro de 2021

Para a realização da avaliação foi realizado o desenvolvimento de um questionário para que os usuários descrevessem como foi a experiências dos mesmo com o jogo.

Em seguida foi realizado um teste com usuários, onde foi apresentados para eles as regras do jogo e disponibilizado o jogo para que eles pudessem jogar. Esse teste foi realizado com estudantes dos cursos ofertados pela universidade Federal do Ceará - Campus Quixadá, que tem em sua estrutura curricular as disciplinas de engenharia de software e/ou requisitos de software. Os estudantes utilizaram o jogo até ter classificados todos os requisitos.

Logo após a utilização do jogo, foi repassado um questionário, onde os participantes do teste responderam sobre sua experiência com o jogo, sobre os pontos positivos e negativos, e, por fim, sobre o seu aprendizado a respeito da classificação de requisitos com a utilização do jogo.

## **5 O JOGO: CLASSIFIQUI**

Neste capítulo são apresentados os pontos referentes ao *game design*, onde na Seção 5.1 é a apresentado o público alvo do jogo, Na 5.2 os conhecimentos prévios necessários, Na 5.3 os objetivos educacionais do jogo, Na 5.4 a descrição do jogo, Na 5.5 o gênero do jogo, Na 5.6 a mecânica (regras) do jogo, Na 5.7 a plataforma do jogo, Na 5.8 a narrativa do jogo, Na 5.9 as formas de interação com o jogo e Na 5.10 as formas de pontuação para definir o jogador campeã.

### **5.1 Público-Alvo**

O público-alvo do jogo são estudantes de cursos de graduação em computação, que conta com a oferta das disciplinas de Engenharia de Software e/ou Requisitos de Software em sua estrutura curricular.

### **5.2 Conhecimento Prévio**

Antes de se utilizar do jogo para auxiliar no aprendizado sobre a classificação de requisitos, os jogadores deverão adquirir antes conhecimento sobre os tipos de requisitos definidos por Wiegers e Beatty (2013), para que assim eles possam desenvolver conhecimento sobre a classificação dos requisitos por meio do jogo.

### **5.3 Objetivos Educacionais**

Tendo em vista que a utilização do jogo busca auxiliar na aprendizagem sobre classificação de requisitos de software, o objetivo educacional do jogo é dispor deste conteúdo de forma que os estudantes aprendam jogando sobre a classificação de requisitos baseadas nas definições de tipos de requisitos de software de Wiegers e Beatty (2013). No Apêndice A é apresentado como foi realizada essa escolha do autor e os tipos de requisitos.

### **5.4 Descrição do Jogo**

O objetivo do jogo é a realização da classificação dos requisitos de um sistema, aplicando conhecimentos sobre as definições dos tipos de requisitos propostos por Wiegers. Para a realização do jogo um dos jogadores criará uma sala no jogo, e assim os demais participantes poderão entrar nessa sala.

## 5.5 Gênero do Jogo

O gênero do jogo é principalmente cartas, onde o mesmo será disposto para os jogadores cartas dos requisitos, tipos de requisitos e bônus.

## 5.6 Mecânica do Jogo

Segundo Brathwaite e Schreiber (2009), a mecânica é a regra que age sobre os jogadores, estado do jogo e visualizações de jogo, e descrevem todas as maneiras de alterar o estado do jogo. A seguir são apresentadas os 5.6.1 Jogadores, as 5.6.2 Cartas e as 5.6.3 Regras.

### 5.6.1 Jogadores

1. A cada partida do jogo participarão no máximo sete jogadores, e no mínimo três jogadores.

### 5.6.2 Cartas

1. Cada jogador terá disponível no mínimo a chance de utilizar uma vez a carta de **analistas de requisitos sênior**, Calazans *et al.* (2017) fala em seu trabalho que o analista de requisitos é a ponte entre as necessidades do cliente e a equipe de desenvolvimento, baseado nessa informação a carta do analista tem o poder de revelar para os jogadores o tipo do requisito para o qual ela foi solicitada.
2. Terão também a chance de utilizar a priori duas vezes a carta de **analistas de requisitos júnior**, os analistas de requisitos júnior possuem a mesma definição de analista de requisitos sênior, porém os estes possuem menos experiência.
3. Terão disponível também uma carta para cada **tipo de requisito**, podendo utilizá-las a quantidade de vezes que for necessário.

### 5.6.3 Regras

1. Para definir qual jogador iniciará a classificação será levado em consideração a ordem de acesso à sala da partida em disputa, ou seja, o primeiro que entrar na sala será o primeiro a classificar e assim sucessivamente.
2. Pensando em classificação de requisitos, o importante é que todos os requisitos do projeto sejam classificados, sendo assim, a quantidade de partidas do jogo será o número de

requisitos que necessitam ser classificados.

3. Será exibido para cada jogador na sua vez de jogar uma carta do *deck* de desafios, depois de ler atentamente a esta carta o jogador poderá realizar a classificação do mesmo. A ordem de exibição destas cartas será a ordem de acesso à sala do jogo.
4. Caso o jogador classifique o requisito de forma correta, este acrescenta 1(um) ponto a sua pontuação e terá o direito de classificar um segundo requisito se este for classificado corretamente o jogador poderá sortear uma carta do *deck* de bônus, após sortear o bônus a jogada passa para o próximo jogador. Caso contrário o requisito passa para o próximo jogador.
5. Será exibido para os jogadores na sua vez de jogar o tipo escolhido pelo jogador anterior caso este erre a classificação.
6. Cada jogador poderá usar as cartas de tipo de requisitos quantas vezes desejarem para classificar os requisitos, isto é, durante a partida cada jogador terá a oportunidade de utilizar cada carta de tipo de requisito quantas vezes achar necessário,, pois estas não possuirão número mínimo ou máximo de utilizações.
7. Cada jogador poderá usar a carta de analistas de requisitos júnior duas vezes a priori, a mesma tem a função de mostrar para o usuário três tipos possíveis para o requisito puxado.
8. Cada jogador poderá usar a carta analista de requisitos sênior a priori uma única vez, onde será possível solicitar ver qual o tipo do requisito.
9. Cada jogador poderá usar a carta analista de requisitos júnior a priori duas vez, onde será possível solicitar ver três o tipos possíveis do requisito.
10. Caso os jogadores utilizem alguma das cartas de ajuda, este não terá o direito de classificar um outro requisito na rodada sendo assim não poderá puxar uma carta do *deck* de bônus.
11. Os jogadores poderão sair do jogo quando todos os requisitos do projeto forem classificados.
12. O jogo termina quando todos os requisitos forem classificados.

## 5.7 Plataforma do Jogo

O jogo trata-se de um jogo de cartas, digital desenvolvido utilizando o *React Native*, este será utilizado para o desenvolvimento do *frontend* e para o *backend* e armazenamento de dados será utilizado o firebase. Sendo assim o jogo utilizará de tecnologias digitais e das cartas como principal ferramenta de interação. Além do mais o jogo será desenvolvido apenas na versão

mobile. O código do jogo desenvolvido poderá ser encontrado em Classifiquei<sup>1</sup>, o aplicativo pode ser baixado em Classifiquei<sup>2</sup>.

## 5.8 Narrativa do Jogo

Chega à empresa um conjunto de requisitos elicitados pelo analista de um sistema, sendo assim é necessário realizar a classificação dos mesmos, para isso são alocados analistas de requisitos e programadores para classificá-los. O processo da classificação se realizado de maneira incorreta, será necessário retrabalho no futuro. Sendo assim, os jogadores deveram manter o projeto dentro do orçamento estabelecido, no contexto do jogo orçamento está relacionado com o tempo de produção do sistema, ou seja, ele tem um prazo para ser desenvolvido e os jogadores deveram manter o projeto nesse prazo.

## 5.9 Formas de Interação com o Jogo

O jogo será composto por quatro *decks*, sendo eles três baseados nos desenvolvidos por Beppe *et al.* (2018) juntamente com um novo que é o *deck* de ajuda.

### 5.9.1 Deck de Desafios

Nesta seção, serão apresentadas as cartas desenvolvidas para serem utilizadas como os desafios do jogo, neste caso os desafios serão os requisitos de software que necessitam serem classificados. Sendo assim, serão construídas as cartas de requisitos de software para quatro sistema selecionados pelo autor, são eles Adote, GREAT Print, GREATOUR, CAFETERIA. O GREAT Print e o GREATOUR foram desenvolvidos pelo GREAt<sup>3</sup>, no qual ele é um laboratório de pesquisa voltado para três áreas de pesquisa são elas redes de computadores, engenharia de software e sistemas, Adote foi um sistema idealizado por Rainara Maia, Cafeteria é um sistema presente no livro do Wieggers e Beatty (2013).

#### 5.9.1.1 Adote

A seguir é apresentada uma descrição para o projeto: É um sistema para apoiar o cuidado e adoção de animais. Centraliza as burocracias de adoção entre quem cuida dos animais

<sup>1</sup> <https://github.com/eleMonteiro/classifiquei>

<sup>2</sup> <https://www.dropbox.com/s/q44sdtqirriyhk/app-release.apk?dl=0>

<sup>3</sup> <https://www.great.ufc.br/>, acesso em: 10 de Fevereiro de 2021

resgatados e quem adota, como também quem ajuda com os custos dos cuidados.

Figura 5 – Cartas do Sistema Adote.

<p><b>O apadrinhamento deve ser feito por cartão de crédito (visa/master)</b></p> 	<p><b>No caso de adoção, uma análise do interessado é realizada por responsáveis da ONG</b></p> 	<p><b>Editar informações</b></p> 	<p><b>Deletar animal</b></p> 
<p><b>Cancelar apadrinhamento</b></p> 	<p><b>O sistema deve requisitar documentação do usuário que deseja adotar ou apadrinhamento</b></p> 	<p><b>Cadastrar animal</b></p> 	<p><b>O sistema deve possibilitar a busca por filtros</b></p> 
<p><b>O sistema deve mostrar uma lista de opções para apadrinhamento</b></p> 	<p><b>Procurar animais</b></p> 	<p><b>A aplicação deve possibilitar a postagem de uma adoção ou apadrinhamento no facebook</b></p> 	<p><b>O usuário deve ser capaz de adotar em 3 cliques</b></p> 
<p><b>O cadastro pode ser feito via google ou facebook</b></p> 	<p><b>A aplicação deve suportar 1000 usuários ao mesmo tempo em dias da semana</b></p> 		

Fonte: Elaborado pelo autor.

### 5.9.1.2 GREAT Print

A seguir é apresentada uma descrição: É uma aplicação que foi desenvolvida para apoiar a impressão de documentos no GREat, para os funcionários dos diversos setores, como do financeiro. Com esse aplicativo os membros do GREat também podem imprimir a partir de seus smartphones e essa impressão deve ser feita na impressora mais próxima do usuário requisitante.

Figura 6 – Cartas do Sistema GREAT Print.

<p><b>Cada funcionário no Great têm direito a no máximo 60 impressões</b></p> 	<p><b>Imprimir arquivo</b></p> 	<p><b>Configurar propriedades de impressão</b></p> 	<p><b>O aplicativo deve atualizar o histórico de impressões no sistema Great</b></p> 
<p><b>Selecionar arquivo para enviar para impressão</b></p> 	<p><b>Cancelar a impressão se esta ainda estiver na fila</b></p> 	<p><b>O aplicativo deve imprimir o arquivo na impressora mais próxima do usuário</b></p> 	<p><b>O aplicativo deve verificar o limite de impressão do usuário</b></p> 
<p><b>O aplicativo deve possuir login com usuário e senha da rede Great</b></p> 	<p><b>O aplicativo deve suportar um total de 100 usuários requisitando ao mesmo tempo</b></p> 	<p><b>O aplicativo deve permitir que o usuário imprima um arquivo com no máximo 2 toques</b></p> 	

Fonte: Elaborado pelo autor.

### 5.9.1.3 GREATOUR

A seguir é apresentada uma descrição para o projeto: É um aplicativo desenvolvido no GREat para apoiar os visitantes do laboratório, que funciona como um guia móvel de visita.

Mostra ao usuário sua localização, as salas, informações da sala, descrição dos funcionários ou pesquisadores e o acompanha em tempo real pelo seu tour de visita no laboratório.

Figura 7 – Cartas do Sistema GREATOUR.

<p><b>Artigos publicados em bases de pesquisa (IEEE, ACM) não podem ser disponibilizados</b></p> 	<p><b>Visualizar textos sobre o local</b></p> 	<p><b>Ativar a adaptação de acordo com a mídia</b></p> 	<p><b>Escutar áudios sobre um local</b></p> 
<p><b>Favoritar vídeo</b></p> 	<p><b>Visualizar local atual dentro do mapa</b></p> 	<p><b>O sistema deve exibir nome, áreas de pesquisa, orientação, letters</b></p> 	<p><b>Visualizar vídeo</b></p> 
<p><b>Visualizar lista de pessoas em um local</b></p> 	<p><b>O sistema deve atualizar as informações quando o local do usuário é atualizado</b></p> 	<p><b>O sistema deve exibir texto, áudio e vídeo se a bateria estiver acima de 50% e a adaptação for ativada</b></p> 	<p><b>O aplicativo deve ser desenvolvido na plataforma android</b></p> 
<p><b>O sistema deve permitir que visitantes visualizem somente informações não confidenciais</b></p> 	<p><b>O sistema deve exibir texto e áudios se a bateria estiver entre 30% e 50%</b></p> 	<p><b>O aplicativo deve utilizar um leitor de PDF para exibir textos</b></p> 	<p><b>O sistema deve reiniciar a interação de onde parou, se o usuário trocar de dispositivo</b></p> 



Fonte: Elaborado pelo autor.

#### 5.9.1.4 Cafeteria

A seguir é apresentada uma descrição para o projeto (WIEGERS; BEATTY, 2013): É um sistema retirado do livro base da disciplina de Requisitos de Software. Serve para apoiar os funcionários de uma empresa gerenciando seus pedidos de comida.

Figura 8 – Cartas do Sistema CAFETERIA.



<p><b>O sistema deve enviar um e-mail ou sms para o usuário após o pedido realizado</b></p> 	<p><b>Criar Menu</b></p> 	<p><b>Deletar Menu</b></p> 	<p><b>Cancelar uma refeição (se ainda não foi preparada)</b></p> 
<p><b>O usuário deve ser capaz de ver a lista e todos os ingredientes do menu</b></p> 	<p><b>Favoritar uma refeição</b></p> 	<p><b>Favoritar uma refeição</b></p> 	<p><b>Pedir uma refeição</b></p> 
<p><b>Alterar a refeição pedida (se ainda não foi preparada)</b></p> 	<p><b>O sistema deve permitir que o usuário recupere um pedido anterior com apenas 1 interação</b></p> 	<p><b>O sistema deve ficar disponível no mínimo 98% do tempo entre 5:00 am e meia noite</b></p> 	<p><b>95% dos usuários devem ser capazes de fazer um pedido na 1ª interação</b></p> 
<p><b>Todas as transações de pagamento pela internet devem usar o algoritmo X.</b></p> 	<p><b>Alterar a refeição pedida (se ainda não foi preparada)</b></p> 	<p><b>O sistema deve permitir que o usuário recupere um pedido anterior com apenas 1 interação</b></p> 	<p><b>O sistema deve ficar disponível no mínimo 98% do tempo entre 5:00 am e meia noite</b></p> 
<p><b>95% dos usuários devem ser capazes de fazer um pedido na 1ª interação</b></p> 	<p><b>Todas as transações de pagamento pela internet devem usar o algoritmo X.</b></p> 		

### 5.9.2 Deck de Jogo

Nesta seção, serão apresentadas as cartas desenvolvidas para serem utilizadas para realizar a resolução dos desafios, são elas as cartas que representam os tipos de requisitos estes são (i) Requisitos de Negócio, (ii) Regras de Negócio, (iii) Restrições, (iv) Requisitos de Interface Externa, (v) *Feature*, (vi) Requisitos Funcionais, (vii) Requisitos Não-Funcionais, (viii) Atributos de Qualidade, (ix) Requisitos de Sistema, (x) Requisitos de Usuário.

Figura 9 – Cartas Tipos de Requisitos.



Fonte: Elaborado pelo autor.

### 5.9.3 Deck de Ajuda

Nesta seção, serão apresentadas as cartas de ajuda, são estas as cartas que representa os jogadores que são as cartas de analista de requisitos sênior e a de analista de requisitos júnior, a carta de analista de requisitos será utilizada pelo o jogador que deseja classificar o requisito de forma correta sem necessitar de um grande esforço, a carta programador sugere três tipos possíveis para o requisito. Todos os jogadores inicialmente ganham a mesma quantidade de

cartas de ajuda do jogo.

Figura 10 – Cartas Ajudas.



Fonte: Elaborado pelo autor.

#### 5.9.4 Deck de Bônus

Nesta seção serão apresentadas as cartas de bônus, as cartas bônus contem para os jogadores que classificarem os requisitos corretamente a cada rodada do jogo.

Figura 11 – Cartas Bônus.

<p>Você foi um excelente analista na classificação desse requisito, por este motivo você irá recuperar a sua carta de ajuda "ANALISTA DE REQUISITOS SÊNIOR" caso você já a tenha utilizado.</p>	<p>Você foi um excelente programador na classificação desse requisito, por este motivo você irá recuperar a sua carta de ajuda "ANALISTA DE REQUISITOS JÚNIOR" caso você já a tenha utilizado.</p>	<p>Você foi um excelente analista na classificação desse requisito, por este motivo você irá ganhar essa sua carta de ajuda "ANALISTA DE REQUISITOS SÊNIOR" você poderá utilizá-la a qualquer momento.</p>	<p>Você foi um excelente programador na classificação desse requisito, por este motivo você irá ganhar essa sua carta de ajuda "ANALISTA DE REQUISITOS JÚNIOR" você poderá utilizá-la a qualquer momento.</p>
<p>Você acabou de ganhar mais "UM PONTO". Parabéns!</p>	<p>"O talento vence jogos, mas só o trabalho em equipe ganha campeonatos." [Michael Jordan] Você pode não está jogando em uma equipe mas acredite você é a sua própria equipe, sendo assim, você ganhou "DOIS PONTOS".</p>	<p>Você acabou de ganhar mais "DOIS PONTOS". Parabéns!</p>	<p>Você acabou de ganhar mais "TRÊS PONTOS". Parabéns!</p>
<p>"O que sabemos é uma gota; o que ignoramos é um oceano." [Isaac Newton] Você deve ter aprendido mais uma gota após essa classificação, sendo assim, você ganhou "UM PONTO".</p>	<p>"Se eu vi mais longe, foi por estar sobre ombros de gigantes." [Isaac Newton] Você viu o mais além possível dos ombros, sendo assim, você ganhou "TRÊS PONTOS".</p>	<p>"Deve-se aprender sempre, até mesmo com um inimigo." [Isaac Newton] Você acaba de ganhar mais "UM PONTO".</p>	

Fonte: Elaborado pelo autor.

## 5.10 Sistema de Pontuação

O jogo não irá conter pontuação negativa, sendo assim o sistema de pontuação do jogo, será realizado através da somatória de pontos obtidos pelos participantes pelas classificações corretas e os bônus obtidos, essa pontuação será exibida para os jogadores em formato de *ranking* pois o mesmo instiga os demais jogadores a tentarem melhorar sua classificação assim realizando uma melhor classificação dos requisitos, para obter uma pontuação para esse *ranking* os jogadores que acertaram suas classificações, além de ganharem um ponto por cada requisito classificado corretamente os jogadores poderão sortear e uma carta de bônus e acrescentar seu valor a pontuação do jogador, essas cartas contêm pontuações que variam de 1 (um) a 3 (três) pontos, serão adicionadas algumas cartas de bônus que não contêm pontuação, estas conterão resgates das cartas de jogadores que na aplicação são representadas pela carta de programador e analista de requisitos.

Ao final de cada partida a aplicação irá calcular a pontuação baseada na pontuação obtida por meio da classificação correta dos requisitos mais a pontuação das cartas de bônus e quem obtiver o melhor resultado será o jogador campeão, este será exibido pela aplicação para os demais participantes.

Figura 12 – Tela do Ranking do Jogo.



Fonte: Elaborado pelo autor.

## 6 AVALIAÇÃO DO JOGO

Neste capítulo a avaliação do jogo é descrita. Na Seção 6.2, é descrito como foi planejada a avaliação do jogo. A Seção 6.2 descreve como foi realizada a execução e como foi feita a coleta dos dados da avaliação. Por fim, na Seção 6.3, são descritos os resultados obtidos na avaliação do jogo.

### 6.1 Planejamento da Avaliação

A avaliação foi planejada para ocorrer no ambiente do usuário, em seus próprios celulares, de forma remota.

Após o uso do jogo, foi solicitado aos participantes responderem um questionário, o qual foi baseado em um questionário utilizado por Souza *et al.* (2017) para validar um jogo de auxílio ao ensino de língua portuguesa.

O questionário possui 13 (treze) perguntas para avaliar a experiência do participantes com o jogo, além de conter perguntas ligadas ao conteúdo disposto no jogo. O questionário de avaliação encontra-se disponível no Apêndice C.

### 6.2 Execução da Avaliação

A aplicação do jogo e a coleta dos dados ocorreram no período de 30 de março de 2021, com 7 (sete) alunos dos cursos de Engenharia de Software e Sistemas de Informação da Universidade Federal do Ceará - Campus Quixadá.

Foi realizado uma reunião através da ferramenta de comunicação do Google, o Google Meet. Após todos os participantes entrarem na chamada do Google Meet, foi pedido aos participantes que eles baixassem o jogo acessando o link[1]<sup>1</sup>, depois que todos baixaram o jogo foi repassado para eles algumas informações a respeito do jogo: (i) as regras do jogo; e (ii) os dados de acesso ao jogo, tendo em vista que o ambiente foi todo configurado para os participantes se preocuparem em apenas entrar no jogo e jogar.

Após todos os jogadores classificarem todos os requisitos de suas respectivas salas, salas estas que disponibilizavam para os usuários os requisitos do sistema adote apresentado no Capítulo 5 na Seção 5.9, foi repassado para eles o link do formulário para que eles respondessem ao questionário.

---

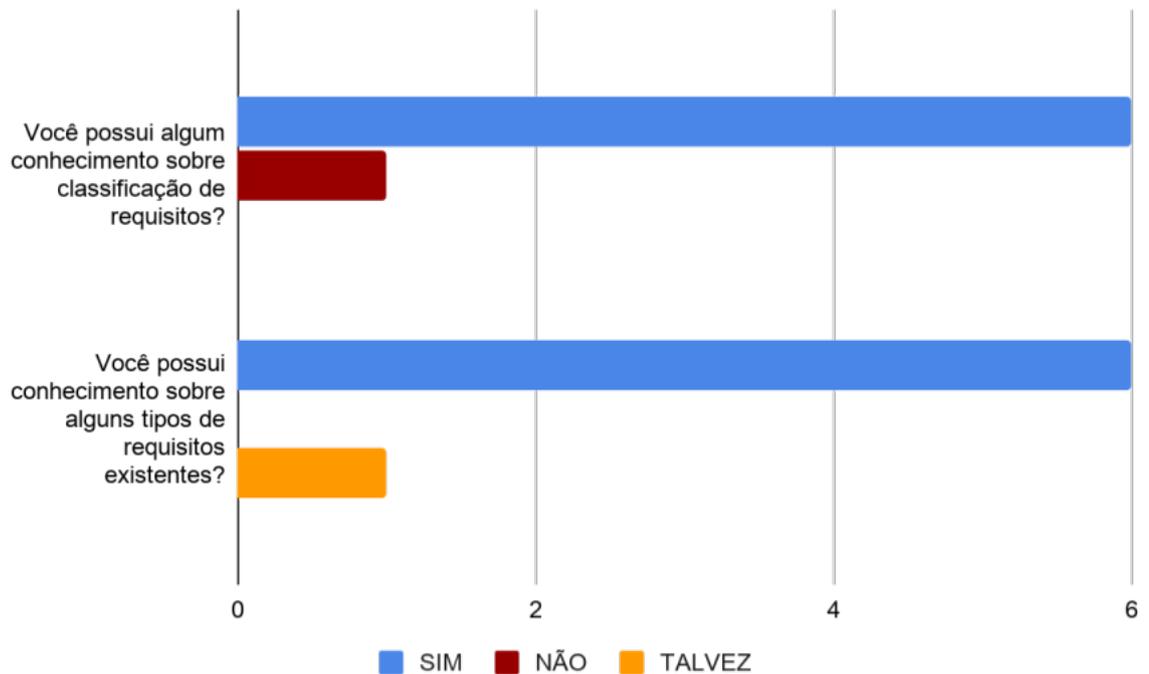
<sup>1</sup> <https://bit.ly/3dnbJ7B>

### 6.3 Resultados da Avaliação

Com os dados coletados, foi possível gerar gráficos e tabelas para apresentação dos resultados. As perguntas da avaliação foram divididas em três dimensões. A primeira é a respeito do conhecimento prévio dos participantes. A segunda é sobre a experiência do usuário e a terceira sobre melhorias ou defeitos do jogo.

A Figura 13 mostra que a maioria dos participantes já realizaram classificação de requisitos e já possuíam conhecimento sobre alguns tipos de requisitos.

Figura 13 – Gráfico do Conhecimento Prévio



Fonte: Elaborado pelo Autor.

Ainda sobre o conhecimento prévio, foi feita uma pergunta subjetiva onde a mesma pode ser encontrada no Quadro 3. Ela trata dos tipos de requisitos conhecidos pelos participantes. Das respostas obtidas, a maioria respondeu tipos de requisitos utilizados no jogo, onde estes foram obtidos a partir do mapeamento feito na literatura e que está descrito no Apêndice A.

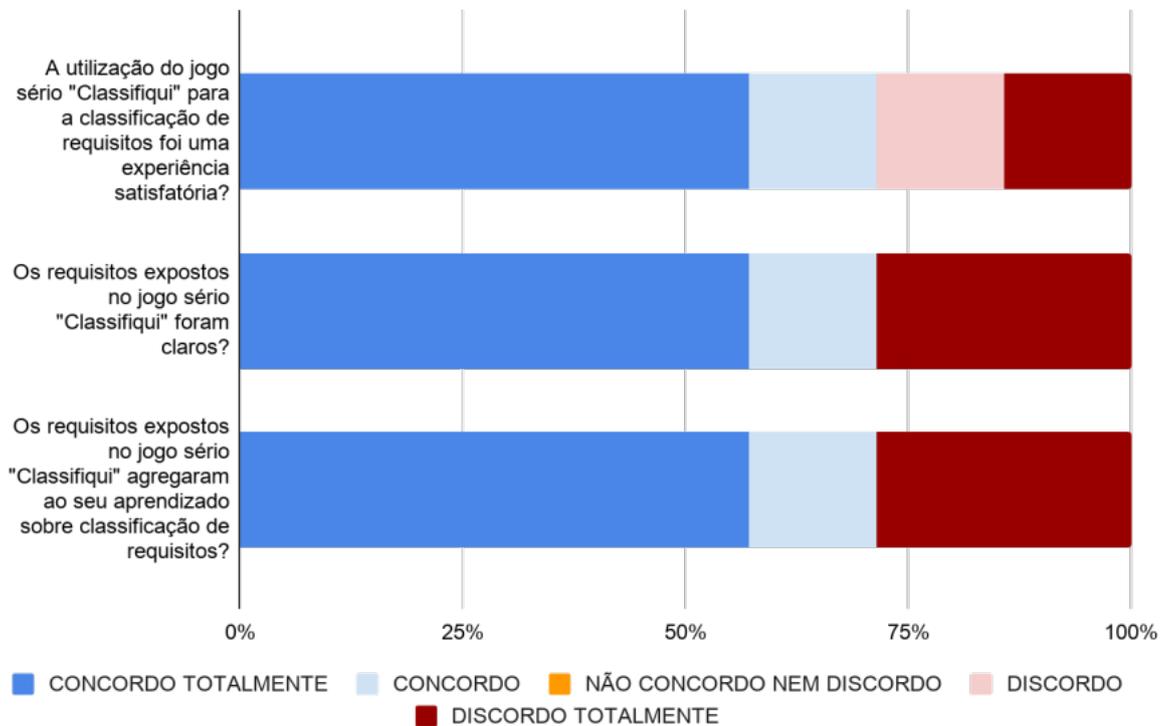
### Quadro 3 – Tipos de Requisitos Conhecidos dos Participantes

<b>Se a sua resposta da pergunta anterior, onde a pergunta anterior é 2 (dois) da Figura 13, foi SIM quais os tipos de requisitos que você conhece?</b>
Requisitos Funcionais, Requisitos Não-funcionais, Regra De Negócio, Requisitos De Usuário, Requisitos De Sistema, Restrições, Requisito De Interface Externa, <i>Feature</i> , Requisitos De Negócio.

Fonte: Elaborado pelo Autor.

Na dimensão Experiência de Usuário, foram feitas 3 (três) perguntas utilizando a escala *Likert* sobre a utilização do jogo, as perguntas e seus resultados podem ser vistos na Figura 14.

Figura 14 – Gráfico da Experiência do Usuário - Parte 1



Fonte: Elaborado pelo Autor.

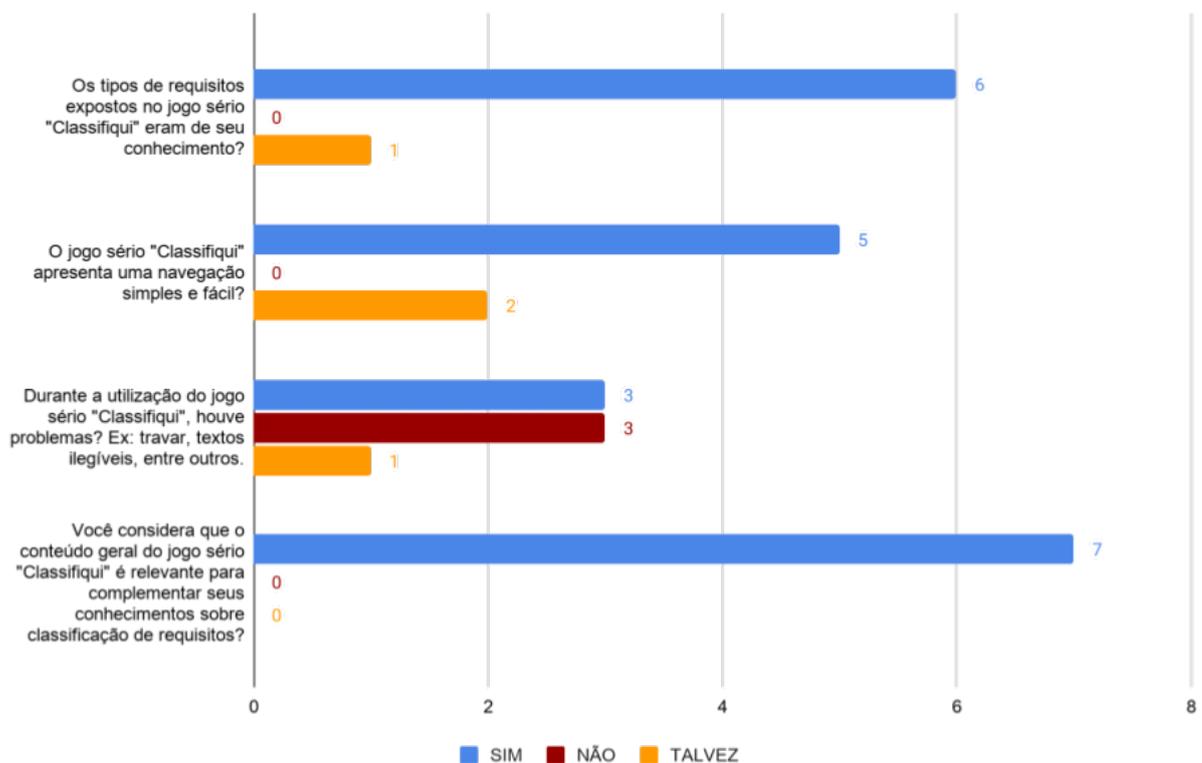
Na primeira pergunta, é perceptível que a maioria dos participantes achou satisfatório a utilização do jogo. Na segunda pergunta, os usuários em sua maioria classificaram o conteúdo do jogo como claro, ou seja, o conteúdo do jogo baseado em seus conhecimentos prévios não deixa dúvidas que impossibilite a classificação dos requisitos.

Já na terceira pergunta, alguns acharam que o conteúdo disposto no jogo não agregou

algo novo ao seu conhecimento. Porém, é importante ressaltar que os requisitos apresentados no jogo já eram de conhecimento dos participantes, pois além de terem sido utilizados em sala de aula durante a disciplina de Requisitos de Software, muitos estão presentes no livro dos autores Wiegers e Beatty (2013).

A Figura 15 apresenta os resultados das perguntas que as respostas eram de sim e não.

Figura 15 – Gráfico da Experiência do Usuário - Parte 2



Fonte: Elaborado pelo Autor.

Na primeira pergunta, a maioria respondeu que os tipos de requisitos apresentados pelo jogo aos participantes eram de conhecimento de 5 (cinco) dos 7 (sete) participantes da coleta de dados.

Pela segunda pergunta, podemos perceber que o jogo apresenta uma navegação simples onde este não é um problema para que os participantes realizassem a classificação dos requisitos.

Na terceira pergunta, percebemos que muitos participantes relataram problemas com

o jogo. O Quadro 4 apresenta alguns dos problemas descritos pelos participantes durante a utilização do jogo no teste com usuários. A maior parte dos problemas estão ligados a questão de usabilidade do jogo e não ao conteúdo apresentado pelo mesmo.

Quadro 4 – Problemas Encontrados pelos Participantes

<b>Se sua resposta da pergunta anterior, onde a pergunta anterior é 4 (quatro) da Figura 15, foi SIM fale um pouco sobre quais os problemas apresentados a vocês pelo jogo sério?</b>
Houve momentos onde a pergunta mesmo sendo acertada por um outro participante voltava para a tela para o participante seguinte avaliar e na primeira tentativa de jogar o jogo Classifiquei não estava pegando as respostas das classificações realizadas pelos participantes.
Uma vez um jogador ficou em loop mesmo errando só ele jogava. Uma vez uma carta foi classificada corretamente mas se repetiu na rodada seguinte.

Fonte: Elaborado pelo Autor.

Por fim, a última pergunta se tratava do conhecimento adquirido pelo participante. Podemos observar que dos 7 (sete) participantes todos os 7 (sete) consideraram que o jogo seria uma boa alternativa para complementar os seus conhecimentos a respeito da classificação de requisitos.

O Quadro 5 descreve as principais dificuldades encontradas pelos participantes no decorrer da utilização do jogo durante o teste com usuários.

Quadro 5 – Dificuldades ou Desvantagens Encontrados pelos Participantes

<b>Sob sua análise, quais as principais dificuldades ou desvantagens foram encontradas ao utilizar o jogo sério "Classifiquei"?</b>
Não ter entendido como funciona o jogo, inicialmente. Mas, após entender, foi tranquilo.
Saber quem era o jogador da vez, houve um pouco de confusão para saber o que fazer em algumas telas.

Fonte: Elaborado pelo Autor.

O Quadro 6 apresenta dados importantes para a evolução do jogo pois nele são apresentadas sugestões de melhoria. Melhorias estas descritas pelos participantes do teste.

Quadro 6 – Sugestões Dadas Pelos Participantes

<b>Quais suas sugestões para possíveis melhorias neste no jogo sério "Classifiquei"?</b>
Seria interessante a implementação de logout.
Era legal que avisasse na que esta na vez de outro jogador, porque no primeiro contato parece que o jogo travou.
Dizer quem acertou a questão da rodada.
Poderia mostrar já no começo em vez de digitar o nome da sala aparecer em uma lista o nome das salas já criadas.
Limitar o tempo de classificação. Dar feedback de quem é o jogador da vez. Colocar alguns botões informativos ou instruções em algumas telas, principalmente na de bônus. Utilizar mais ícones para uma melhor experiência de usuário.

Fonte: Elaborado pelo Autor.

Por fim, esses dados coletados forneceram evidências de que o jogo pode vir a produzir um efeito positivo no ensino da classificação de requisitos.

## 7 CONCLUSÃO

Através de uma revisão da literatura, foi possível observar a necessidade do ensino da classificação de requisitos de software de forma mais divertida.

Dessa forma, esse trabalho focou no desenvolvimento de um jogo sério digital para apoiar o ensino de requisitos no que diz respeito aos seus tipos, tais como Requisito Funcional, Não-Funcional, Regra de Negócio, entre outros.

O jogo Classifiquei, apresentado como a principal contribuição deste trabalho foi desenvolvido baseado inteiramente no conteúdo abordado nos primeiros capítulos deste trabalho. Este possibilita ao jogador colocar em prática o que foi aprendido anteriormente de forma teórica sobre classificação de requisitos. O material do jogo é um aplicativo onde suas telas podem ser observadas no Apêndice B, e seu código está disponível em Classifiquei<sup>1</sup>.

Após a criação do jogo, foi realizado um teste com usuários a fim de obter dados relacionados a experiência do participantes com o jogo, além de relacionados ao conteúdo disposto no jogo. Este teste com usuários buscava validar a utilização do jogo para o que ele é proposto.

Com os resultados obtidos, foi possível observar que o jogo pode facilitar o ensino da classificação de requisitos de software.

Um ponto que pode afetar as conclusões desta pesquisa é o fato de não terem sido dispostos no jogo requisitos para todos os tipos apresentados neste trabalho sendo assim algo que pode e deve ser melhorado futuramente é isto.

Atualmente o Classifiquei só conta com requisitos de três tipos, além de estar disponível apenas para sistemas Android. Como trabalho futuro é possível desenvolver requisitos para todos os tipos descritos neste trabalho além de ser desenvolvido para sistemas IOS, e melhorar o design do jogo a partir das possibilidades de melhorias identificadas na sua aplicação.

---

<sup>1</sup> <https://github.com/eleMonteiro/classifiquei>

## REFERÊNCIAS

- AURUM, A.; WOHLIN, C. **Engineering and managing software requirements**. [S. l.]: Springer Science & Business Media, 2005.
- AYDAN, U.; YILMAZ, M.; CLARKE, P. M.; O'CONNOR, R. V. Teaching iso/iec 12207 software lifecycle processes: A serious game approach. **Computer Standards Interfaces**, [S. l.]: Elsevier, v. 54, p. 129–138, 2017.
- BEPPE, T. A.; ARAÚJO, Í. L. de; ARAGÃO, B. S.; SANTOS, I. de S.; XIMENES, D.; ANDRADE, R. M. C. Greatest: a card game to motivate the software testing learning. In: **Proceedings of the XXXII Brazilian Symposium on Software Engineering**. [S. l.: s. n.], 2018. p. 298–307.
- BRATHWAITE, B.; SCHREIBER. **Challenges for Game Designers**. [S. l.]: Course Technology, 2009. ISBN 9781584505808.
- BRITO, A. de S. **'2TSCRUM'**: um jogo de tabuleiro para o ensino do Scrum. Quixadá, Brasil, 2016. Disponível em: <http://www.repositorio.ufc.br/handle/riufc/24946>. Acesso em: 14 abr. 2020.
- CALAZANS, A.; PALDÊS, R.; BRAOSI, E.; REZENDE, K.; PEREIRA, N. O perfil do analista de requisitos de software: uma comparação entre a academia e o mercado de trabalho brasileiro. In: **WER17 - Workshop em Engenharia de Requisitos**. Buenos Aires, Argentina: [S. n.], 2017.
- COLPANI, R.; FARIA, M. Joy e as letrinhas: um serious game como ferramenta de auxílio no processo de alfabetização de crianças do ensino fundamental. [S. l.]: **Revista Brasileira de Informática na Educação**, v. 25, n. 02, p. 61, 2017.
- GARCÍA, I.; PACHECO, C.; LEÓN, A.; CALVO-MANZANO, J. A. Experiences of using a game for improving learning in software requirements elicitation. **Computer Applications in Engineering Education**, Wiley Online Library, v. 27, n. 1, p. 249–265, 2019.
- GARCÍA, I.; PACHECO, C.; LEÓN, A.; CALVO-MANZANO, J. A. A serious game for teaching the fundamentals of iso/iec/ieee 29148 systems and software engineering–lifecycle processes–requirements engineering at undergraduate level [S. l.]: **Computer Standards & Interfaces**, Elsevier, v. 67, p. 103–377, 2020.
- GONÇALVES, R. Q.; THIRY, M.; ZOUCCAS, A. Avaliação da aprendizagem em experimentos com jogo educativo de engenharia de requisitos. [S. l.]: **X Simpósio Brasileiro de Qualidade de Software (SBQS), Curitiba**, 2011.
- HAINES, T.; CONNOLLY, T. M.; STANSFIELD, M.; BOYLE, E. A. Evaluation of a game to teach requirements collection and analysis in software engineering at tertiary education level. [S. l.]: **Computers & Education**, Elsevier, v. 56, n. 1, p. 21–35, 2011.
- ISO/IEC. **ISO 29148**: ISO/IEC/IEEE International Standard - Systems and software engineering – Life cycle processes – Requirements engineering. Geneva, 2018.
- LAUESEN, S. **Software requirements**: styles and techniques. Pearson Education, 2002.

LIMA, T.; CAMPOS, B.; SANTOS, R.; WERNER, C. Ubire: A game for teaching requirements in the context of ubiquitous systems. In: IEEE. **2012 XXXVIII Conferencia Latinoamericana En Informatica (CLEI)**. [S. l.], 2012. p. 1–10.

MEC. **Diretrizes Curriculares Nacionais elaboradas pela Comissão de Especialistas de Ensino da área da Computação**. [S. l.], 2016. Disponível em: <http://www.mec.gov.br>. Acesso em: 19 abr. 2020.

MICHAEL, D.; CHEN, S. **Serious Games: Games That Educate, Train, and Inform**. Thomson, Canadá, 2006.

MICHAELIS, C. **Dicionário Escolar de Língua Portuguesa**. [S. l.]: Melhoramentos, 2016. v. 4. ISBN 978-85-06-04024-9.

NAVARRO-ALMANZA, R.; JUUREZ-RAMIREZ, R.; LICEA, G. Towards supporting software engineering using deep learning: A case of software requirements classification. **Proceedings - 2017 5th International Conference in Software Engineering Research and Innovation**, [S. l.]: CONISOFT 2017, v. 2018-January, p. 116–120, 2018.

PRATES, R. O.; BARBOSA, S. D. J. Avaliação de interfaces de usuário—conceitos e métodos. **Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação**, [S. l.]: v. 6, 2003.

PRENSKY, M. **Digital Game-Based Learning**. [S. l.]: McGraw-Hill Pub. Co., 2004. ISBN 0071454004.

PRESSMAN, R.; MAXIM, B. **Engenharia de Software - 7ª Edição**. [S. l.]: McGraw Hill Brasil, 2011. ISBN 9780073375977.

ROCHA, R. V. da; ARAUJO, R. B. de. Metodologia de design de jogos sérios para treinamento. [S. l.]: **SBC – Proceedings of SBGames**, 2013.

SAKUDA, L.; FORTIM, I. **II Censo da Indústria Brasileira de Jogos Digitais**. Brasília: Ministério da Cultura, 2018.

SAVI, R.; ULBRICHT, V. R. Jogos digitais educacionais: benefícios e desafios. [S. l.]: **RENTE-Revista Novas Tecnologias na Educação**, v. 6, n. 1, 2008.

SAVI, R.; WANGENHEIM, C. G. von; BORGATTO, A. F. A model for the evaluation of educational games for teaching software engineering. In: IEEE. **2011 25th Brazilian Symposium on Software Engineering**. [S. l.], 2011. p. 194–203.

SILVA A., C. **Jogo educacional para apoiar o ensino de técnicas para elaboração de testes de unidade**. Dissertação (Mestrado) – Universidade do Vale do Itajaí, São José, 2010. Dissertação de Curso de Mestrado, Computação Aplicada, UNIVALI.

SILVA J., C. E. d. **Investigação do ensino de engenharia de requisitos na perspectiva da academia e da indústria: um enfoque em documentação de requisitos**. Natal, 2018. Dissertação de Mestrado em Sistemas e Computação- Centro de Ciências Exatas e da Terra, UFRN.

SOARES, M. dos S. **Projeto de Jogos Educativos 2D de Aventura usando Lua**. Tese (Doutorado) – PUC Rio, Rio de Janeiro, 2012.

SOMMERVILLE, I. **Engenharia de Software**. São Paulo: Pearson, 2011. v. 9. ISBN 978-85-7936-108-1.

SOUZA, P. H. d. S.; LAURINDO, A. K. S. *et al.* **Aplicativos educacionais**: um estudo de caso no desenvolvimento de um aplicativo na plataforma App Inventor2 para auxílio no ensino de produção textual nas aulas de português. Araranguá: [S. n.], 2017.

WIEGERS, K.; BEATTY, J. **Software Requirements, Third Edition**. Washington: Microsoft Press, 2013. ISBN 978-0-7356-7966-5.

YASIN, A.; LIU, L.; LI, T.; WANG, J.; ZOWGHI, D. Design and preliminary evaluation of a cyber security requirements education game (sreg). [S. l.]: **Information and Software Technology**, Elsevier, v. 95, p. 179–200, 2018.

## APÊNDICE A – MAPEAMENTO DOS TIPOS DE REQUISITOS

O mapeamento dos requisitos se torna essencial pois será nessa etapa que é definido o conteúdo a ser utilizado no jogo. Sendo assim, para mapear os tipos de requisitos a serem utilizados, serão desenvolvidas três atividades. Após isso, será possível iniciar a criação do *game design*. A seguir são apresentadas as três atividades necessárias para a realização do mapeamento.

1. **Buscar classificações de tipos de requisitos:** Será realizada uma busca na literatura, a fim, de encontrar classificações propostas para os requisitos, feitas por diferentes autores.
2. **Comparar classificações seguindo critérios definidos:** Para escolher quais classificações utilizar, serão definidos critérios de exclusão de classificações, para que seja possível chegar a um grupo menor de classificações.
3. **Escolher uma classificação para a construção do jogo:** Depois de realizada as atividades anteriores, as classificações que restarem depois de passar por todos os critérios de exclusão serão escolhidas para serem utilizadas como o conteúdo do jogo.

### 1. Sommerville (2011)

- a) Requisitos Funcionais: São ações ou funções que o sistema deve realizar.
- b) Requisitos Não-Funcionais: São restrições de ações ou funções que o sistema deve fornecer.
  - i. Requisitos de Produto: Requisitos que especificam ações ou restrições do sistema.
  - ii. Requisitos Externos: Requisitos que derivam de fatores externos ao sistema, como leis governamentais, entre outros.
  - iii. Requisitos Organizacionais: Requisitos derivados de especificações organizacionais do ambiente para o qual o sistema está sendo desenvolvido.

### 2. Wiegers e Beatty (2013)

- a) Requisitos de Negócio: São objetivos que a empresa deseja obter a partir da utilização do sistema.
- b) Regras de Negócio: São leis ou regras da empresa no qual o software deverá obedecer.
- c) Restrições: São definições que os desenvolvedores deverão seguir para a construção do software. Por exemplo linguagem de programação a ser utilizada, banco de dados.
- d) Requisitos de Interface Externa: São sistemas de software ou hardware externos ao sistema a ser desenvolvido no qual o mesmo se comunica.

- e) *Features*: São recursos do sistema ligado a sua lógica que agregam valor para o usuário.
- f) Requisitos Funcionais: São ações ou funções que o sistema deverá realizar.
- g) Requisitos Não-Funcionais: São propriedades que o sistema deverá apresentar e respeitar sob algumas circunstâncias de uso.
- h) Atributos de Qualidade: São tipos específicos de requisitos não-funcionais que descreve um serviço ou característica de desempenho de um produto.
- i) Requisitos de Sistema: São requisitos nível acima dos requisitos de produto que contém subsistemas, que podem ser todos software ou software e hardware.
- j) Requisitos de Usuário: São metas ou tarefas que classes específicas de usuários devem poder executar com um sistema ou com um atributo de produto desejado.

### 3. Pressman e Maxim (2011)

- a) Requisitos Normais: São objetivos e metas estabelecidas com o cliente no qual deverão ser obtidos. Por exemplo: funções específicas que o sistema realizará.
- b) Requisitos Esperados: São requisitos implícitos ao cliente mas que se existirem deixarão o cliente mais satisfeito. Por exemplo: facilidade na interação homem-máquina, confiabilidade, entre outros.
- c) Requisitos Fascinantes: São recursos vão além da expectativa dos clientes e demonstram ser muito satisfatórios quando presentes.

### 4. Aurum e Wohlin (2005)

- a) Requisitos Organizacionais: São os objetivos e estratégias que a empresa deseja garantir ou obter com o uso do sistema.
- b) Requisito de Produto: São geralmente requisitos funcionais e não funcionais. Onde os mesmos deverão garantir que os requisitos estejam alinhados com as metas e objetivos em termos de produto.
- c) Requisitos de Projeto: São requisitos que definem como será realizado o desenvolvimento do sistema. Por exemplo: planejamento, gerenciamento de riscos, orçamento e custo.

### 5. Lauesen (2002)

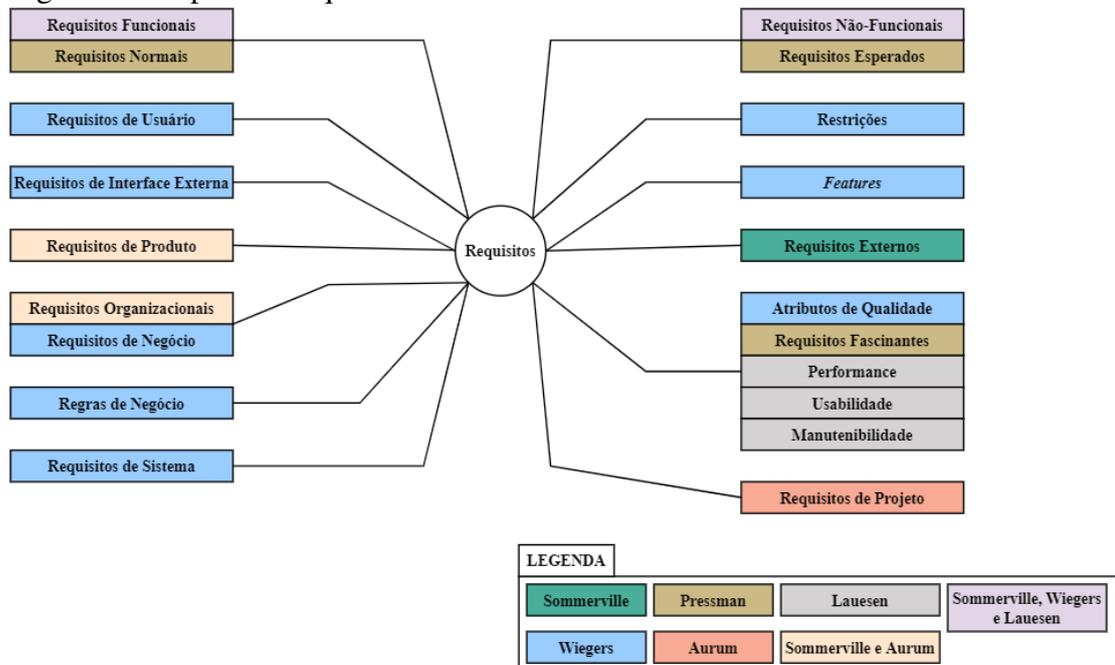
- a) Requisitos Funcionais: São funções que o sistema realizará por meio de entradas do usuário.
- b) Requisitos Não-Funcionais: São especificações do quão bem o sistema executa suas

funções pretendidas. Os mesmos também são chamados de requisitos de qualidade.

- i. Performance: Representa o quão eficiente o sistema realiza suas funções e comunicações com o hardware.
- ii. Usabilidade: Representa quão eficiente e fácil é a interação do usuário com o sistema.
- iii. Manutenibilidade: Representa quão fácil deve ser para reparar defeitos, como é fácil adicionar novas funcionalidades.

Tendo em vista as definições propostas pelos autores, é possível perceber semelhanças entre as definições e os tipos de requisitos propostos pelos mesmos. Sendo assim a Figura 16, apresenta a união dos tipos de requisitos semelhantes entre os apresentados pelos autores.

Figura 16 – Tipos de Requisitos



Fonte: Elaborado pelo Autor.

Baseado nos dados apresentados na Figura 16, foram definidos critérios para selecionar qual autor será utilizado para definir os tipos de requisitos a serem utilizados no presente trabalho. Os critérios são:

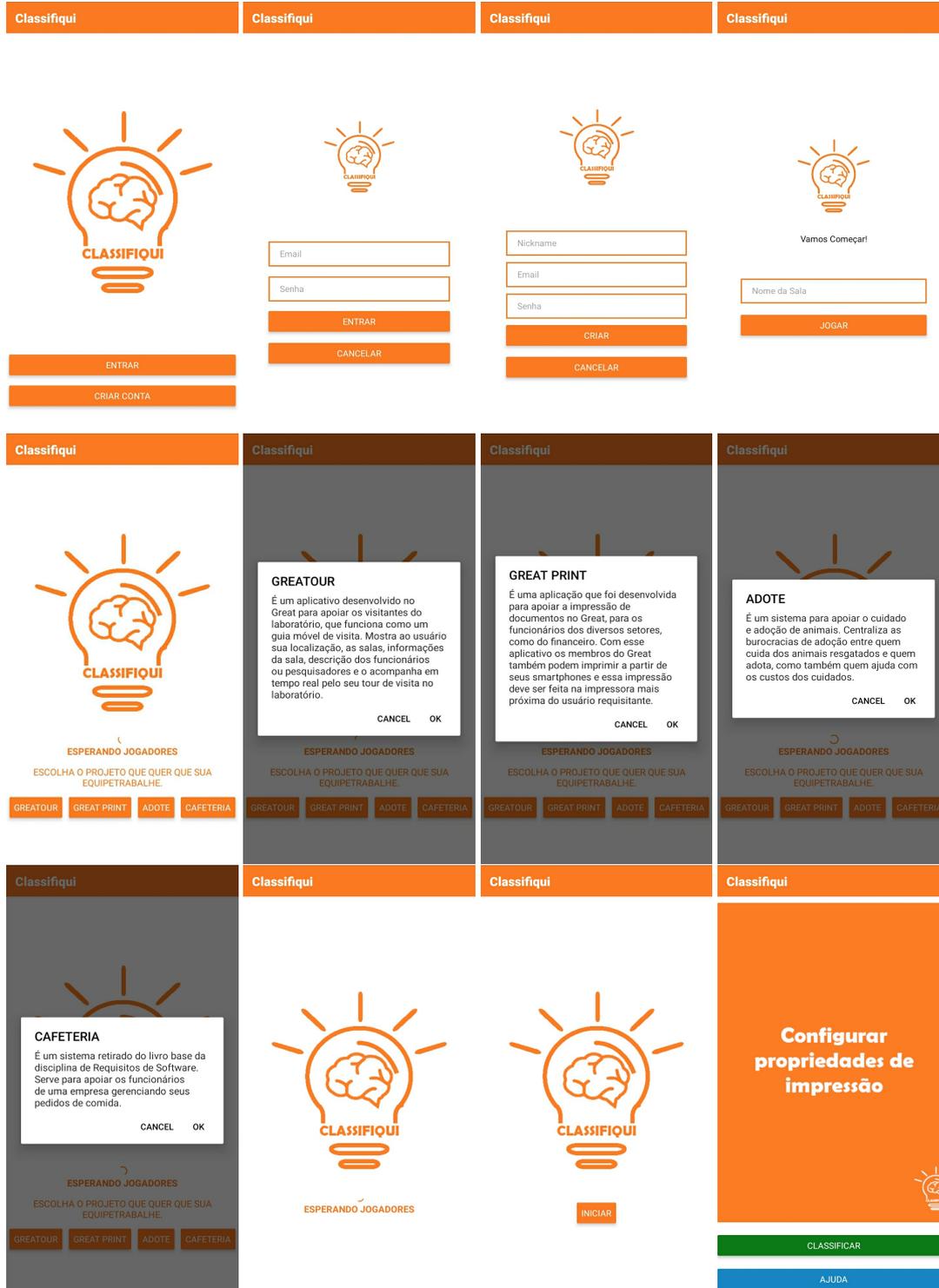
- (i) Se a obra do autor está presente na Bibliografia utilizada nas ementas das disciplinas de Requisitos de Software e Engenharia de Software ofertadas pela Universidade Federal do Ceará - Campus de Quixadá;
  - Os autores obtidos como resultado para este critério foram: **Sommerville, Wieggers e Pressman.**

(ii) Variação de tipos de requisitos.

- O autor que melhor representa este critério é **Wieggers**, pois pela Imagem 16 pode-se perceber que ele apresenta uma variação maior para os tipos de requisitos.

## APÊNDICE B – TELAS DO APLICATIVO DO JOGO

Figura 17 – Telas do Aplicativo do Jogo.





Fonte: Elaborado pelo autor.

## APÊNDICE C – QUESTIONÁRIO DE AVALIAÇÃO DO JOGO

Todos os dados serão coletados anonimamente e somente serão utilizados no contexto desta pesquisa.

Pesquisador: Francisco Elenilson Pessoa Monteiro – elemonteiro@alu.ufc.br Universidade Federal do Ceará – Campus Quixadá

1. Qual o seu curso?
  - b. Ciência da Computação
  - d. Engenharia de Software
  - e. Sistemas de Informação
2. Tem em sua grade curricular as disciplinas?
  - a. Engenharia de Software
  - b. Requisitos de Software
  - c. Ambas as Disciplinas
  - d. Nenhuma das Disciplinas
3. Você possui algum conhecimento sobre classificação de requisitos?
  - a. Sim
  - b. Não
4. Você possui conhecimento sobre alguns tipos de requisitos existentes?
  - a. Sim
  - b. Não
5. Se a sua resposta da pergunta anterior foi SIM quais os tipos de requisitos que você conhece?
6. A utilização do jogo sério "Classifique" para a classificação de requisitos foi uma experiência satisfatória?
  - a. Concordo Totalmente
  - b. Concordo
  - c. Não concordo nem discordo
  - d. Discordo
  - e. Discordo Totalmente
7. Os requisitos expostos no jogo sério "Classifique" foram claros?
  - a. Concordo Totalmente
  - b. Concordo

- c. Não concordo nem discordo
  - d. Discordo
  - e. Discordo Totalmente
8. Os requisitos expostos no jogo sério "Classifiquei" agregaram ao seu aprendizado sobre classificação de requisitos?
- a. Concordo Totalmente
  - b. Concordo
  - c. Não concordo nem discordo
  - d. Discordo
  - e. Discordo Totalmente
9. Os tipos de requisitos expostos no jogo sério "Classifiquei" eram de seu conhecimento?
- a. Sim
  - b. Não
10. O jogo sério "Classifiquei" apresenta uma navegação simples e fácil?
- a. Sim
  - b. Não
11. Durante a utilização do jogo sério "Classifiquei", houve problemas? Ex: travar, textos ilegíveis, entre outros
- a. Sim
  - b. Não
12. Se sua resposta da pergunta anterior foi SIM fale um pouco sobre quais os problemas apresentados a vocês pelo jogo sério?
13. Você considera que o conteúdo geral do jogo sério "Classifiquei" é relevante para complementar seus conhecimentos sobre classificação de requisitos?
- a. Sim
  - b. Não
14. Sob sua análise, quais as principais dificuldades ou desvantagens foram encontradas ao utilizar o jogo sério "Classifiquei"?
15. Quais suas sugestões para possíveis melhorias neste no jogo sério "Classifiquei"?

## APÊNDICE D – UTILIZAÇÃO DO JOGO

1. **Inicialmente os usuários irão acessar o link**<sup>1</sup>
2. **Deverão baixar e instalar o apk do jogo em seu *smartphone***
3. **Em seguida executar o apk.** A Figura 18 apresenta a primeira tela quando o jogo é executado.

Figura 18 – Tela Inicial do Jogo.



Fonte: Elaborado pelo autor.

4. **Criar uma conta.** A Figura 19 apresenta a tela onde o usuário irá criar a sua conta no jogo.

<sup>1</sup> <https://www.dropbox.com/s/q44sdtqirriyhk/app-release.apk?dl=0>

Figura 19 – Tela de Criar Conta no Jogo.



Classifiqui



Formulário de criação de conta:

- Input: Nickname
- Input: Email
- Input: Senha
- Botão: CRIAR
- Botão: CANCELAR

Fonte: Elaborado pelo autor.

5. **Fazer login.** A Figura 20 apresenta a tela onde o usuário irá realizar o login na sua conta do jogo.

Figura 20 – Tela de Login do Jogo.



Classifiqui



Formulário de login:

- Input: Email
- Input: Senha
- Botão: ENTRAR
- Botão: CANCELAR

Fonte: Elaborado pelo autor.

6. **Criar uma sala.** A Figura 21 apresenta a tela onde o usuário irá criar a sala na qual irá acessar no jogo.

Figura 21 – Tela de Criar Sala no Jogo.



Fonte: Elaborado pelo autor.

7. **Selecionar o sistema dentre os disposto no jogo.** A Figura 22 apresenta a tela onde o usuário irá selecionar o projeto no qual realizará a classificação de seus requisitos.

Figura 22 – Tela de Selecionar o Sistema no Jogo.



Fonte: Elaborado pelo autor.

8. **Convidar amigos para jogar junto.** A Figura 23 apresenta a tela onde o usuário irá ficar esperando os demais jogadores acessar a sala.

Figura 23 – Tela Esperando Amigos.



Fonte: Elaborado pelo autor.

9. **Iniciar a classificação.** A Figura 24 apresenta a tela onde os usuários irão iniciar a classificação dos requisitos.

Figura 24 – Tela Inicial da Classificação.



Fonte: Elaborado pelo autor.