

Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia de Software

Avaliação e melhoria de jogos para aprendizagem de IHC: implementando mecanismos de usabilidade

**Autores: João Pedro Soares Cirqueira e Moacir Mascarenha
Soares Junior**

Orientador: Dr. André Barros de Sales

Brasília, DF

2022



João Pedro Soares Cirqueira e Moacir Mascarenha Soares Junior

Avaliação e melhoria de jogos para aprendizagem de IHC: implementando mecanismos de usabilidade

Monografia submetida ao curso de graduação em (Engenharia de Software) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia de Software).

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Dr. André Barros de Sales

Coorientador: Dra. Fabiana Freitas Mendes

Brasília, DF

2022

João Pedro Soares Cirqueira e Moacir Mascarenha Soares Junior

Avaliação e melhoria de jogos para aprendizagem de IHC: implementando mecanismos de usabilidade / João Pedro Soares Cirqueira e Moacir Mascarenha Soares Junior. – Brasília, DF, 2022-

145 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr. André Barros de Sales

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB
Faculdade UnB Gama - FGA , 2022.

1. Usabilidade. 2. Mecanismos de usabilidade. Jogos para aprendizagem. I. Dr. André Barros de Sales. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Avaliação e melhoria de jogos para aprendizagem de IHC: implementando mecanismos de usabilidade

CDU 02:141:005.6

João Pedro Soares Cirqueira e Moacir Mascarenha Soares Junior

Avaliação e melhoria de jogos para aprendizagem de IHC: implementando mecanismos de usabilidade

Monografia submetida ao curso de graduação em (Engenharia de Software) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia de Software).

Trabalho aprovado. Brasília, DF, 18 de Novembro de 2022:

Dr. André Barros de Sales
Orientador

Dra. Fabiana Freitas Mendes
Coorientadora

Dra. Milene Serrano
Convidada 1

**Me. Eduardo Gabriel Queiroz
Palmeira**
Convidado 2

Brasília, DF
2022

Agradecimentos

Eu, João Pedro Soares Cirqueira, agradeço aos meus pais, Adriana e Maurício, por sempre me apoiarem e incentivarem nos estudos. Agradeço aos meus familiares por sempre apoiar minhas decisões. Agradeço minha namorada Bruna pelo apoio e compreensão. Agradeço aos professores da graduação por compartilharem seus conhecimentos, em especial ao orientador, Dr. André Barros de Sales e coorientadora Dra. Fabiana Freitas Mendes. Por fim, agradeço a todos os colegas de curso pelas experiências compartilhadas.

Eu, Moacir Mascarenha Soares Junior, agradeço primeiramente aos meus pais, Moacir e Adenice, por todo apoio e incentivo. Agradeço a minha irmã Raquel pelo suporte e colaboração. Agradeço a todos os professores, em especial ao orientador Dr. André Barros de Sales e coorientadora Dra. Fabiana Freitas Mendes, pela dedicação e ensino. Por fim, agradeço meus amigos e colegas que contribuíram para a realização deste trabalho.

Agradecemos aos criadores dos jogos, PersonaDesignGame e Universidade das Heurísticas, Rossicler Pires júnior e Cauê Oliveira e Lucas Naves pela disponibilidade e documentação disponibilizada. Também agradecemos aos alunos Ailamar Guimarães e Heron Sousa pela colaboração na avaliação do jogo PDG.

Resumo

A usabilidade é importante para a qualidade de um software, pois está relacionada com a facilidade do usuário em interagir com o software, melhorando assim a experiência do usuário. A usabilidade no contexto da engenharia de software possui características e seus subtipos, nomeados como mecanismo de usabilidade. Este trabalho tem como principal objetivo avaliar a usabilidade de dois jogos em relação aos mecanismos de usabilidade e implementar melhorias. A pesquisa foi dividida em quatro fases, sendo elas: (i) elaboração do projeto de pesquisa, (ii) entendimento dos jogos, (iii) coleta e avaliação, e (iv) proposta e implementações de melhorias. Neste trabalho, as fases i, ii, iii, iv foram executadas para os jogos PersonaDesignGame (PDG) e Universidade das Heurísticas (UnH). Os resultados obtidos, com a execução das fases, foram: o grau de implementação dos mecanismos de usabilidade e a lista de problemas encontrados, sendo assim, possível planejar e realizar melhorias na usabilidade dos jogos.

Palavras-chaves: Usabilidade. Mecanismos de usabilidade. Jogos para aprendizagem.

Abstract

Usability is important for the quality of software, as it is related to the how easy is to interact with the software and can improve the user's experience. Usability in the context of software engineering has characteristics and its subtypes, named as usability mechanism. This work aims to evaluate the usability of two games in relation to a set of usability mechanisms and implementation improvements. The research was divided into four phases, namely: (i) elaboration of the research project, (ii) understanding the games, (iii) collection and evaluation and (iv) Improvement proposal and implementations. In this work phases i, ii, iii, iv were performed for the games PersonaDesignGame (PDG) and Universidade das Heuristics (UnH). The results obtained, with the execution of the phases, are: the degree of implementation of the usability mechanisms, a list of problems found, being possible to plan and carry out improvements in the usability of the games.

Key-words: Usability. Usability mechanisms. Games for learning.

Lista de ilustrações

Figura 1 – EAP do projeto.	28
Figura 2 – Cronograma do projeto em TCC1.	30
Figura 3 – Cronograma do projeto em TCC2.	31
Figura 4 – Universidade das Heurísticas.	38
Figura 5 – PersonaDesignGame.	39
Figura 6 – Fases da pesquisa.	43
Figura 7 – Coleta e consolidação dos resultados.	50
Figura 8 – Protótipo botões PDG	76
Figura 9 – Protótipo botões UnH	77
Figura 10 – Protótipo alerta interação PDG	78
Figura 11 – Protótipo alerta interação UnH	78
Figura 12 – Protótipo alerta de confirmação PDG	79
Figura 13 – Protótipo alerta de confirmação UnH	79
Figura 14 – Diagrama de classe resposta dos serviços	80
Figura 15 – Diagrama padronização retorno serviço para <i>controller</i>	80
Figura 16 – Interface de padronização dos erros UnH	81
Figura 17 – Protótipo alerta de falha no sistema PDG	81
Figura 18 – Protótipo alerta de falha no sistema UnH	81
Figura 19 – Protótipo carregamento sistema.	82
Figura 20 – Protótipo descrição entrada de texto	82
Figura 21 – Protótipo alerta erro em campo de texto	82
Figura 22 – PDG - Interação sem melhorias	85
Figura 23 – PDG - Interação com melhorias	86
Figura 24 – UnH - Interação sem melhorias	86
Figura 25 – UnH - Interação com melhorias	87
Figura 26 – PDG - Aviso com melhorias	88
Figura 27 – UnH - Aviso com melhorias	88
Figura 28 – PDG - Estado do sistema antes da melhoria	89
Figura 29 – PDG - Estado do sistema com melhoria	90
Figura 30 – UnH - Estado do sistema com melhoria	90
Figura 31 – UnH - Entrada de texto estruturada com melhoria	92
Figura 32 – UnH - Execução passo a passo sem melhoria	93
Figura 33 – UnH - Execução passo a passo com melhoria	93
Figura 34 – Falha no sistema.	143
Figura 35 – Ação com o mouse.	144
Figura 36 – Falta de descrição da consequência da ação.	144

Figura 37 – Tela projetos UnH.	144
Figura 38 – Tela cadastro usuário UnH.	145
Figura 39 – Instruções para questionário UnH.	145

Lista de tabelas

Tabela 1 – Mecanismos de usabilidade	36
Tabela 3 – Estrutura caso de uso	46
Tabela 4 – Estrutura padrão para apoio da avaliação.	47
Tabela 5 – Estrutura apoio inspeção.	48
Tabela 6 – Estrutura padrão identificação de violação.	48
Tabela 7 – UC01-Responder questões	54
Tabela 8 – UC02-Visualizar conteúdo	55
Tabela 9 – UC03-Visualizar recompensas	55
Tabela 10 – Revisão em dupla-PDG	56
Tabela 11 – Primeira revisão em grupo	57
Tabela 12 – Consenso primeira revisão	59
Tabela 13 – Segunda revisão em grupo	60
Tabela 13 – Segunda revisão em grupo	61
Tabela 14 – Consenso segunda revisão	61
Tabela 15 – Resultado revisão em grupo-PDG	62
Tabela 16 – UC01- Cadastrar usuário	65
Tabela 17 – UC02- Responder questões	66
Tabela 18 – UC03- Verificar <i>ranking</i>	66
Tabela 19 – UC04- Informações sobre o jogo	66
Tabela 20 – Primeira revisão em dupla	67
Tabela 20 – Primeira revisão em dupla	68
Tabela 20 – Primeira revisão em dupla	69
Tabela 20 – Primeira revisão em dupla	70
Tabela 21 – Consenso primeira revisão	70
Tabela 22 – Revisão em dupla - UnH	71
Tabela 23 – <i>Feedback</i> : Estado do sistema	105
Tabela 24 – <i>Feedback</i> : Interação	106
Tabela 25 – <i>Feedback</i> : <i>Feedback</i> para ação longa	106
Tabela 26 – <i>Feedback</i> : Aviso	106
Tabela 27 – Desfazer/Cancelar: Desfazer global	107
Tabela 28 – Desfazer/Cancelar: Desfazer específico do objeto	107
Tabela 29 – Desfazer/Cancelar: Abortar Operação	107
Tabela 30 – Desfazer/Cancelar: Voltar	108
Tabela 31 – Prevenção/correção de erros de entrada do usuário: Entrada de Texto Estruturado	108
Tabela 32 – Assistente: Execução Passo a Passo	108

Tabela 33 – Perfil de Usuário: Preferências	109
Tabela 34 – Perfil de Usuário: Espaço de Objetos Pessoais	109
Tabela 35 – Perfil de Usuário: Favoritos	109
Tabela 36 – Ajuda: Ajuda Multinível	110
Tabela 37 – Agregação de Comandos	110
Tabela 38 – Inspeção - Estado do sistema	111
Tabela 39 – Violação - Estado do sistema	111
Tabela 40 – Inspeção - Interação	111
Tabela 41 – Violação - Interação	112
Tabela 42 – Inspeção - <i>Feedback</i> para ações longas	112
Tabela 43 – Inspeção - Aviso	112
Tabela 44 – Violação - Aviso	113
Tabela 45 – Inspeção - Desfazer global	113
Tabela 46 – Inspeção - Desfazer específico do objeto.	113
Tabela 47 – Violação - Desfazer específico do objeto	114
Tabela 48 – Inspeção - Abortar Operação	114
Tabela 49 – Violação - Abortar Operação	114
Tabela 50 – Inspeção - Voltar	115
Tabela 51 – Violação - Voltar	115
Tabela 52 – Inspeção - Preferências	115
Tabela 53 – Violação - Preferências	116
Tabela 54 – Inspeção - Espaço de Objetos Pessoais	116
Tabela 55 – Violação - Espaço de Objetos Pessoais	116
Tabela 56 – Inspeção - Favoritos	116
Tabela 57 – Violação - Favoritos	117
Tabela 58 – Inspeção - Ajuda multinível	117
Tabela 59 – Violação - Ajuda multinível	117
Tabela 60 – Inspeção - Agregação de Comandos	117
Tabela 61 – Inspeção - Execução Passo a Passo	118
Tabela 62 – Inspeção - Entrada de texto estruturado	118
Tabela 63 – Inspeção - Estado do sistema	119
Tabela 64 – Violação - Estado do sistema	119
Tabela 65 – Inspeção - Interação	119
Tabela 66 – Violação - Interação	120
Tabela 67 – Inspeção - <i>Feedback</i> para ações longas	120
Tabela 68 – Inspeção - Aviso	120
Tabela 69 – Violação - Aviso	121
Tabela 70 – Inspeção - Desfazer global	121
Tabela 71 – Violação - Desfazer global	121

Tabela 72 – Violação - Desfazer específico do objeto	122
Tabela 73 – Inspeção - Desfazer específico do objeto.	122
Tabela 74 – Violação - Desfazer específico do objeto	122
Tabela 75 – Inspeção - Abortar Operação	123
Tabela 76 – Violação - Abortar Operação	123
Tabela 77 – Inspeção - Voltar	123
Tabela 78 – Violação - Voltar	124
Tabela 79 – Inspeção - Prevenção/correção de erros de entrada de usuário	124
Tabela 80 – Violação - Prevenção/correção de erros de entrada de usuário	124
Tabela 81 – Inspeção - Passo a Passo	124
Tabela 82 – Inspeção - Preferências	125
Tabela 83 – Violação - Preferências	125
Tabela 84 – Inspeção - Espaço de Objetos Pessoais	125
Tabela 85 – Violação - Espaço de Objetos Pessoais	125
Tabela 86 – Inspeção - Favoritos	126
Tabela 87 – Inspeção - Ajuda multinível	126
Tabela 88 – Violação - Ajuda multinível	126
Tabela 89 – Inspeção - Agregação de Comandos	126
Tabela 90 – Inspeção - Estado do sistema	127
Tabela 91 – Violação - Estado do sistema	127
Tabela 93 – Violação - Interação	127
Tabela 92 – Inspeção - Interação	128
Tabela 94 – Inspeção - <i>Feedback</i> para ações longas	128
Tabela 95 – Inspeção - Aviso	128
Tabela 96 – Violação - Aviso	129
Tabela 97 – Inspeção - Desfazer global	129
Tabela 98 – Inspeção - Desfazer específico do objeto.	129
Tabela 99 – Inspeção - Abortar Operação	130
Tabela 100 – Violação - Abortar Operação	130
Tabela 101 – Inspeção - Voltar	130
Tabela 102 – Inspeção - Prevenção/correção de erros de entrada de usuário	131
Tabela 103 – Violação - Prevenção/correção de erros de entrada de usuário	131
Tabela 104 – Inspeção - Passo a Passo	131
Tabela 105 – Violação - Passo a Passo	131
Tabela 106 – Inspeção - Preferências	132
Tabela 107 – Inspeção - Espaço de Objetos Pessoais	132
Tabela 108 – Inspeção - Favoritos	132
Tabela 109 – Inspeção - Ajuda multinível	133
Tabela 110 – Violação - Ajuda multinível	133

Tabela 111–Inspeção - Agregação de Comandos	133
Tabela 112–Inspeção - Estado do sistema	134
Tabela 113–Violação - Estado do sistema	134
Tabela 114–Inspeção - Interação	134
Tabela 115–Violação - Interação	135
Tabela 116–Inspeção - <i>Feedback</i> para ações longas	135
Tabela 117–Inspeção - Aviso	135
Tabela 118–Violação - Aviso	136
Tabela 119–Inspeção - Desfazer global	136
Tabela 120–Inspeção - Desfazer específico do objeto.	136
Tabela 121–Inspeção - Abortar Operação	137
Tabela 122–Violação - Abortar Operação	137
Tabela 123–Inspeção - Voltar	137
Tabela 124–Inspeção - Prevenção/correção de erros de entrada de usuário	138
Tabela 125–Violação - Prevenção/correção de erros de entrada de usuário	138
Tabela 126–Inspeção - Passo a Passo	138
Tabela 127–Violação - Passo a Passo	138
Tabela 128–Inspeção - Preferências	139
Tabela 129–Inspeção - Espaço de Objetos Pessoais	139
Tabela 130–Inspeção - Favoritos	139
Tabela 131–Inspeção - Ajuda multinível	140
Tabela 132–Violação - Ajuda multinível	140
Tabela 133–Inspeção - Agregação de Comandos	140
Tabela 134–Magnitude coeficiente Kappa	143

Lista de abreviaturas e siglas

ABNT	Associação Brasileira de Normas Técnicas
API	Interface de Programação de Aplicações
EAP	Estrutura analítica do projeto
GDD	<i>Game Design Document</i>
IHC	Interação Humano Computador
NBR	Norma Técnica Brasileira
OE	Objetivo Específico
PDG	PersonaDesignGame
PHP	<i>Personal Home Page</i>
TCC	Trabalho de Conclusão de Curso
TDD	<i>Technical Design Document</i>
UC	Caso de Uso
UnB	Universidade de Brasília
UnH	Universidade das Heurísticas

Lista de símbolos

κ - Índice de concordância Kappa.

P_o - Somatório das concordâncias encontradas.

P_e - Somatório concordâncias atribuíveis ao acaso.

Sumário

1	INTRODUÇÃO	25
1.1	Justificativas	26
1.2	Objetivos	26
1.3	Definição de Escopo	26
1.3.1	Estrutura Analítica do Projeto-EAP	27
1.4	Cronograma	29
1.5	Resumo do Capítulo	32
2	REFERENCIAL TEÓRICO	33
2.1	Interação Humano Computador	33
2.2	Mecanismos de Usabilidade	34
2.3	Jogos Educacionais	37
2.3.1	Universidade das Heurísticas	37
2.3.2	PersonaDesignGame	39
2.4	Trabalhos Correlatos	40
2.5	Resumo do Capítulo	40
3	METODOLOGIA	43
3.1	Elaboração do Projeto de Pesquisa	44
3.2	Entendimento dos Jogos	45
3.2.1	Estudo da Documentação dos Jogos	45
3.2.2	Instalação e Execução dos Jogos	45
3.3	Coleta e Avaliação	46
3.4	Melhorias	51
3.5	Resumo do Capítulo	51
4	AVALIAÇÃO DO JOGO PDG	53
4.1	Coleta de Dados	53
4.1.1	Preparação	53
4.2	Avaliação	55
4.2.1	Revisão dos Resultados: Dupla (PDG)	56
4.2.2	Revisão dos Resultados: Grupo (PDG)	57
4.2.2.1	Dificuldade Revisão em Grupo	62
4.2.2.2	Resultado Revisão em Grupo	62
4.3	Resumo do Capítulo	63

5	AVALIAÇÃO DO JOGO UNH	65
5.1	Coleta de Dados	65
5.1.1	Preparação	65
5.2	Avaliação	67
5.2.1	Revisão dos Resultados: Dupla (UnH)	67
5.2.1.1	Resultado Revisão em Dupla	71
5.3	Resumo do Capítulo	71
6	PLANEJAMENTO DAS MELHORIAS NOS JOGOS	73
6.1	Mecanismos Escolhidos	73
6.2	Arquitetando Melhorias	73
6.2.1	Interação	74
6.2.2	Aviso	74
6.2.3	Estado dos Sistema	74
6.2.4	Entrada de Texto Estruturada	75
6.2.5	Execução Passo a Passo	75
6.3	Modelagem	75
6.4	Resumo do Capítulo	83
7	IMPLEMENTAÇÃO DAS MELHORIAS NOS JOGOS	85
7.1	Interação	85
7.2	Aviso	87
7.3	Estado do Sistema	89
7.4	Entrada de Texto Estruturada	91
7.5	Execução Passo a Passo	92
7.6	Condução do Desenvolvimento	94
7.7	Resumo do Capítulo	94
8	CONCLUSÃO	95
8.1	Futuras Evoluções	96
	REFERÊNCIAS	99
	APÊNDICES	103
	APÊNDICE A – MECANISMOS DE USABILIDADE-TABELAS DE APOIO	105
B	– AVALIAÇÃO PDG	111
B.1	Revisão em dupla	111

C	–	AVALIAÇÃO UNH	119
C.1		Avaliação individual: João	119
C.2		Avaliação individual: Moacir	127
C.3		Revisão em dupla	133

ANEXOS **141**

ANEXO A – PRIMEIRO ANEXO	143
---------------------------------	------------

1 Introdução

A usabilidade ¹ é um critério bastante importante para a qualidade de uso, em aplicações, percebida pelos usuários. De acordo com Nielsen (1993), a facilidade no aprendizado e uso de interfaces, além da satisfação ao decorrer do uso, são aspectos relacionados à usabilidade.

Na disciplina Interação Humano Computador (IHC), critérios de qualidades são frequentemente abordados, como a usabilidade, experiência do usuário e comunicabilidade (BARBOSA; JUNQUEIRA, 2010). Deste modo, a melhora nestes aspectos aumenta a qualidade de uso em sistemas interativos, sendo neste trabalho a usabilidade o principal aspecto abordado.

A usabilidade em software é abordada no trabalho de Juristo, Moreno e Sanchez-Segura (2007), no qual as características de usabilidade são decompostas em subtipos nomeados como mecanismos de usabilidade. A implementação destes mecanismos impactam de forma positiva a usabilidade de sistemas interativos.

Os jogos educacionais, segundo Falkembach (2006), apresentam conteúdos e atividades práticas, reunindo lazer e diversão com objetivo principal o aprendizado. Os jogos digitais podem ser sistemas interativos, que cada vez mais, vem demonstrando ser ferramentas com potencial para ensino e aprendizagem.

Segundo Gee (2009), bons jogos (videogames) com foco em aprendizagem podem incorporar alguns princípios, sendo a interação um destes princípios. A interação usuário-sistema é um processo retratado em IHC, que segundo Barbosa e Junqueira (2010), abrange manipulação, comunicação, conversa, troca, influência, dentre outros.

No projeto de pesquisa “Recurso Didático para Interação Humano-Computador” ², foram desenvolvidos jogos com foco na aprendizagem da disciplina de IHC. Este trabalho visa dar continuidade aos jogos, PersonaDesignGame (Seção 2.3) e Universidade das Heurísticas (Seção 2.3), melhorando a usabilidade, com a implementação de alguns dos mecanismos de usabilidade. Portanto, facilitando o aprendizado e uso de interfaces e aumentando a satisfação do usuário com uso do sistema.

¹ Usabilidade - Conjunto de atributos relacionados com o esforço necessário para o uso de um sistema interativo, e relacionados com a avaliação individual de tal uso, por um conjunto específico de usuários (ABNT, 2003). Melhor detalhada na Seção 2.1.

² Recurso Didático para Interação Humano-Computador - É um projeto de pesquisa realizado na Universidade de Brasília, apresentando trabalhos relacionados com a disciplina de Interação Humano Computador.

1.1 Justificativas

Nesta seção são apresentadas as justificativas, motivos principais que levaram à realização deste trabalho.

A usabilidade é um critério importante para a qualidade de uso sistemas interativos. Sua implementação auxilia na aprendizagem e no uso do sistema, contribuindo significativamente para a satisfação do usuário.

Foram encontrados trabalhos, detalhados na Seção 2.4, nos quais foram realizadas melhorias na usabilidade de sistemas, implementando mecanismo de usabilidade nas fases iniciais dos projetos (requisitos). Portanto, foi observada a oportunidade da aplicação dos mecanismos de usabilidade nas etapas finais do ciclo de vida de um *software* (jogos).

Para aplicar estas melhorias, foram selecionados, com auxílio dos orientadores, os jogos PDG e UnH, por se tratarem de jogos educacionais na disciplina de IHC desenvolvidos no projeto de pesquisa “Recurso Didático para Interação Humano-Computador” (UnB). Desta maneira, aplicar melhorias na usabilidade destes jogos é uma ótima oportunidade para a implementação de alguns mecanismos de usabilidade em aplicações que já concluíram a etapa de desenvolvimento.

1.2 Objetivos

O objetivo geral deste trabalho é **melhorar a usabilidade de dois jogos educacionais digitais**. Para alcançar este objetivo, foram propostos os seguintes objetivos específicos:

- **OE01:** Conhecer os conceitos sobre usabilidade, mecanismos de usabilidade e jogos digitais de apoio ao processo de ensino-aprendizagem;
- **OE02:** Identificar necessidades de melhorias na usabilidade em jogos digitais, com foco nos mecanismos de usabilidade;
- **OE03:** Selecionar mecanismos de usabilidade que serão implementados;
- **OE04:** Projetar e implementar melhorias aos jogos para aumentar a usabilidade.

1.3 Definição de Escopo

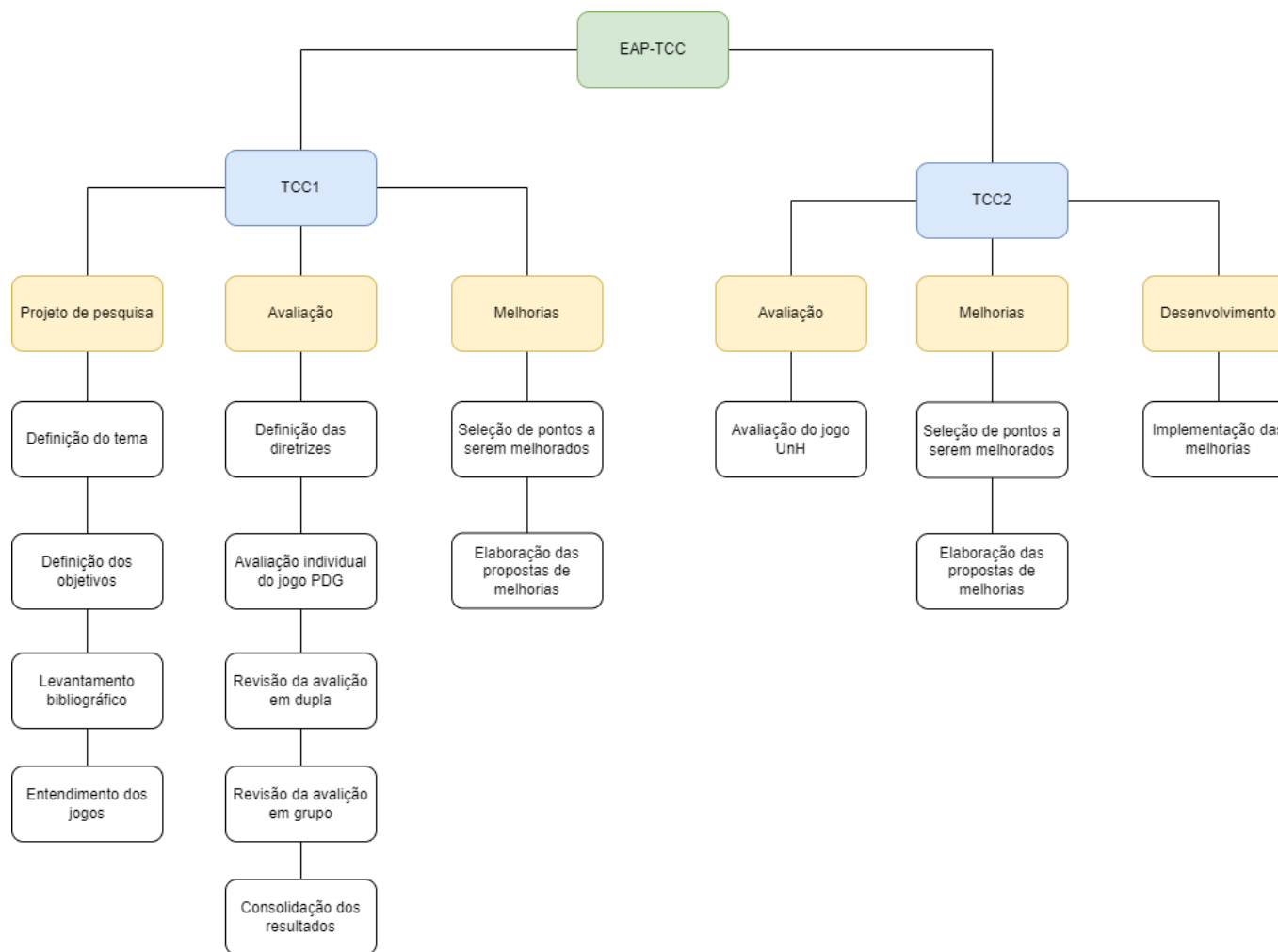
Nesta seção, as atividades e o cronograma são apresentados, relacionando as atividades e as datas delimitadas para o cumprimento das mesmas. Portanto, o projeto seguiu as atividades planejadas.

1.3.1 Estrutura Analítica do Projeto-EAP

Segundo [PMI \(2017\)](#), a EAP é uma decomposição hierárquica do escopo total do trabalho a ser executado pela equipe do projeto a fim de atingir os objetivos do projeto e criar as entregas requeridas.

A Figura 1 representa de forma diagramática a EAP definida para o projeto. As etapas definidas na EAP (Figura 1) foram estabelecidas para que os objetivos sejam alcançados ao final do projeto.

Figura 1 – EAP do projeto.



Fonte: Autores.

1.4 Cronograma

Nesta seção são apresentados os cronogramas com as atividades definidas, que foram executadas durante o período estimado para a conclusão do projeto. As Figuras 2 e 3 apresentam as atividades e seus respectivos períodos de execuções, seguidas pela dupla de autores durante a execução do projeto.

Figura 2 – Cronograma do projeto em TCC1.

Data		Atividades	Semanas														
Início	Fim		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
17/01/22	13/02/22	Definição do cronograma	█	█	█	█											
24/01/22	27/03/22	Estudo trabalhos correlatos		█	█	█	█	█									
24/01/22	13/02/22	Estudo mecanismos de usabilidade		█	█	█											
24/01/22	27/02/22	Estudo "estudo de casos"		█	█	█	█	█									
24/01/22	13/02/22	Estudo interação humano computador		█	█	█											
24/01/22	10/04/22	Estudo outras bibliografias		█	█	█	█	█	█	█	█	█	█	█			
24/01/22	27/03/22	Estrutura TCC		█	█	█	█	█	█	█	█	█					
24/01/22	20/02/22	Definição do escopo		█	█	█	█										
31/01/22	10/04/22	Redação TCC		█	█	█	█	█	█	█	█	█	█	█			
14/02/22	27/02/22	Estudo de Caso: Planejamento					█	█									
28/02/22	27/03/22	Avaliação/Diagnostico/Entendimento do jogo							█	█	█	█					
28/03/22	10/04/22	Elaboração do relatorio sobre o resultado do estudo											█	█			
28/03/22	10/04/22	Prosposta projeto de software: Prototipação											█	█			
14/03/22	10/04/22	Prosposta projeto de software: Diagrama de classes											█	█			
11/04/22	17/04/22	Revisão dos professores														█	
18/04/22	24/04/22	Revisão/Ajustes do TCC															█
24/04/22	01/05/22	Apresentação															█

Fonte: Autores.

Figura 3 – Cronograma do projeto em TCC2.

Data		Atividades	Semanas															
Início	Fim		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
06/06/2022	12/06/2022	Revisão dos mecanismos de usabilidade	█	█														
13/06/2022	26/06/2022	Estudo trabalhos correlatos		█	█													
13/06/2022	03/07/2022	Estudo outras bibliografias		█	█	█												
13/06/2022	14/08/2022	Estrutura TCC		█	█	█	█	█	█	█	█	█						
13/06/2022	28/08/2022	Redação TCC		█	█	█	█	█	█	█	█	█	█	█				
13/06/2022	26/06/2022	Estudo de Caso: Planejamento		█	█													
27/06/2022	17/07/2022	Avaliação/Diagnostico/Entendimento do jogo UnH				█	█	█										
27/06/2022	17/07/2022	Elaboração do relatório sobre o resultado do estudo UnH				█	█	█										
18/07/2022	07/08/2022	Proposta projeto de software: Prototipação UnH							█	█	█							
18/07/2022	07/08/2022	Proposta projeto de software: Diagrama de classes UnH							█	█	█							
08/08/2022	04/09/2022	Implementação das propostas nos jogos PDG e UnH										█	█	█	█			
08/08/2022	04/09/2022	Verificar/Planejar/Implementar: Padrões de Projeto										█	█	█	█			
05/09/2022	11/09/2022	Revisão dos professores														█		
12/09/2022	18/09/2022	Revisão/Ajustes do TCC															█	
19/09/2022	25/09/2022	Apresentação																█

Fonte: Autores.

As atividades definidas nos cronogramas para o TCC1 e TCC2 (Figuras 2 e 3) foram planejadas para todo o período do projeto. Por conta da pandemia, os semestres de realização do projeto foram realizados com um período letivo menor em relação a semestres anteriores à pandemia.

1.5 Resumo do Capítulo

Neste Capítulo foram abordados conceitos base para o projeto de forma introdutória, para um entendimento geral do trabalho. Os conceitos abordados são relacionados com a usabilidade com foco na experiência do usuário e qualidade do produto. Estes critérios, critérios são comumente abordados em IHC.

A partir dos conceitos, foi apresentada a justificativa (Seção 1.1) para o projeto, tendo como foco a melhoria na usabilidade de jogos digitais aplicando mecanismos de usabilidade à aplicações já desenvolvidas. Além disso, foi exposto o objetivo geral e os objetivos específicos (Seção 1.2), definidos para o desenvolvimento do trabalho.

Com as definições do escopo do projeto e as atividades executadas no desenvolvimento do trabalho, foi possível a construção da EAP (Subseção 1.3.1) e o cronograma (Subseção 1.4) para o projeto.

2 Referencial Teórico

Neste capítulo são apresentados conceitos necessários para o entendimento dessa proposta. Na Seção 2.1, conceitos relacionados à IHC, seguido de uma breve revisão de mecanismos de usabilidade (Seção 2.2) são apresentados. Nas seções seguintes, contém uma visão geral do objeto de estudo desse projeto, os jogos Universidade das Heurísticas (Seção 2.3.1) e PersonaDesignGame (Seção 2.3.2). Finalmente, a Seção 2.4 revisa alguns trabalhos similares ao que é desenvolvido nesse projeto.

2.1 Interação Humano Computador

Nesta seção são apresentados conceitos relacionados à área de IHC, como: interação, interface, usabilidade e comunicabilidade. Sendo estes conceitos relevantes para o entendimento e a construção do trabalho.

IHC é uma disciplina interessada no projeto, implementação e avaliação de sistemas computacionais interativos para uso humano, juntamente com os fenômenos relacionados a esse uso (HEWETT et al., 1992). De acordo com Hewett et al. (1992), os objetos de estudo de IHC podem ser agrupados em cinco tópicos inter-relacionados: a natureza da interação humano-computador; o uso de sistemas interativos situado em contexto; características humanas; arquitetura de sistemas computacionais e da interface com usuários; e processos de desenvolvimento preocupados com uso (BARBOSA; JUNQUEIRA, 2010).

De acordo com Norman (1988), a interação indica um processo através do qual o usuário formula uma intenção, planeja suas ações, atua sobre a interface, percebe e interpreta a resposta do sistema, e avalia se seu objetivo foi alcançado. Em geral, a interação usuário-sistema pode ser considerada como tudo o que acontece quando uma pessoa e um sistema computacional se unem para realizar tarefas, visando a um objetivo (HIX; HARTSON, 1993).

A interface de um sistema interativo compreende toda a porção do sistema com a qual o usuário mantém contato físico ou conceitual durante a interação (MORAN, 1981). Ela é o único meio de contato entre o usuário e o sistema. Por isso, a grande maioria dos usuários acredita que o sistema é a interface com a qual entram em contato (HIX; HARTSON, 1993).

A norma técnica brasileira ABNT (2003) define usabilidade, uma característica da qualidade de *software*, como sendo um “conjunto de atributos relacionados com o esforço necessário para o uso de um sistema interativo, e relacionados com a avaliação individual de tal uso, por um conjunto específico de usuários”. Neste trabalho, será utilizada a defini-

ção de usabilidade como característica de qualidade de *software* apresentada pela ABNT (2003).

A comunicabilidade diz respeito à capacidade da interface de comunicar ao usuário a lógica do *design*: as intenções do *designer* e os princípios de interação resultantes das decisões tomadas durante todo o processo de *design* (PRATES; BARBOSA; SOUZA, 2000; SOUZA; BARBOSA, 2005; SOUZA; LEITÃO, 2009). As chances de um uso criativo do usuário é maior quando se entende a lógica utilizada na concepção do sistema interativo.

2.2 Mecanismos de Usabilidade

A usabilidade é um atributo da qualidade de software, onde segundo Juristo, Moreno e Sanchez-Segura (2006), deve ser considerada nas fases iniciais do processo de desenvolvimento, diferentemente das abordagens antigas, onde a usabilidade podia ser tratada tardiamente no processo de desenvolvimento.

No estudo realizado por Juristo, Moreno e Sanchez-Segura (2007), a usabilidade pode ser decomposta em características com o intuito de examinar os impactos na arquitetura de software. Juristo, Moreno e Sanchez-Segura (2007) abordam em seu trabalho, as sete características que apresentam benefícios para a usabilidade, com grandes implicações para o sistema, estas características são descritas a seguir:

- **Feedback:** O usuário deve ser mantido informado sobre o que está acontecendo no sistema;
- **Cancelar/desfazer:** As entradas de dados realizadas pelo usuário, em caso de erro deve-se ter a possibilidade da correção do mesmo ou mecanismos de prevenção.
- **Prevenção/correção de erros de entrada de usuário:** As entradas de dados feitas pelo usuário, devem ser prevenidas ou corrigidas;
- **Assistente:** Em tarefas que envolvem muitos passos, o usuário deve ser auxiliado;
- **Perfil do usuário:** Deve ser possível que usuário adapte a aplicação de acordo com suas preferências;
- **Ajuda:** Ao realizar tarefas, o usuário deve receber ajuda de como realizá-la.;
- **Agregação de comando:** A aplicação deve permitir que o usuário crie comandos para executar várias atividades ao mesmo tempo.

Para cada uma das características, Juristo, Moreno e Sanchez-Segura (2007) identificaram e categorizaram, um conjunto de variantes ou subtipos, os quais foram nomeados

de mecanismos de usabilidade. Essas variantes no estudo de [Juristo, Moreno e Sanchez-Segura \(2007\)](#) são levadas em consideração nas etapas iniciais do processo de desenvolvimento (requisitos). A categorização dos mecanismos de usabilidade e seus objetivos são descritos na Tabela 1.

Tabela 1 – Mecanismos de usabilidade

Características de Usabilidade	Mecanismos de usabilidade	Objetivo
<i>Feedback</i>	Interação	Informar o usuário do sistema que uma interação do usuário foi registrada
	<i>Feedback</i> para ação longa	Informar o usuário sobre uma ação que levará muito tempo para ser completada
	Alerta / Aviso	Informar o usuário sobre ação com importantes consequências
	Estado do sistema	Manter o usuário informado sobre o estado do sistema
Cancelar/Desfazer	Desfazer global	O sistema deve desfazer ações em vários níveis
	Desfazer em objeto específico	Desfazer várias ações em um objeto
	Abortar Operação	Cancelar uma ação específica ou a aplicação
	Voltar	Voltar a um estado específico dentro de uma sequência de execuções
Prevenção/correção de erros de entrada de usuário	Entrada de texto estruturada	Ajudar a prevenir que o usuário cometa erros na entrada de dados
Assistente	Execução Passo-a-Passo	Ajudar usuários a realizar tarefas que contem diferentes passo com entradas do usuário e corrigir tal entrada
Perfil de Usuário	Preferências	Gravar as opções de cada usuário para usar as funções do sistema
	Espaço de objeto pessoal	Gravar as opções de cada usuário para usar a interface do sistema
	Favoritos	Gravar determinados locais de interesse para o usuário
Ajuda	Ajuda multinível	Fornecer diferentes níveis de ajuda para diferentes usuários
Agregação de Comandos	Agregação de Comandos	Expressar possíveis ações a serem tomadas com o software por meio de comandos que podem ser construídos a partir de partes menores

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2007\)](#).

Com as informações sobre os mecanismos de usabilidade indicadas por [Juristo](#),

Moreno e Sanchez-Segura (2007), torna-se possível realizar verificações e avaliações dos requisitos em aplicações já desenvolvidas. Portanto, é viável a definição de soluções que aprimorem a usabilidade do sistema.

2.3 Jogos Educacionais

Os jogos educacionais, segundo Falkembach (2006), apresentam conteúdos e atividades práticas, reunindo lazer e diversão, tendo como objetivo principal o aprendizado.

A tecnologia está cada vez mais presente no dia a dia das pessoas. Prensky (2021) apontou que está acontecendo uma revolução na aprendizagem no século XXI, principalmente por conta da tecnologia. Deste modo, a aprendizagem baseada em jogos digitais vem crescendo em diversos lugares, acarretando na necessidade de atualização do ensino com contribuição da tecnologia, como é apresentado por Prensky (2021).

Segundo Prensky (2021), indivíduos com menos de 40 anos, em sua maioria, cresceram utilizando os jogos como uma maneira de diversão. Com essa nova forma de entretenimento, habilidades e preferências foram moldadas oferecendo às pessoas um grande potencial para aprendizagem. Os jogos para aprendizagem, segundo Sena et al. (2016), precisam se atentar a fatores como a preservação do interesse e a motivação dos estudantes, além de garantir a ligação entre os objetivos instrucionais e a jogabilidade.

Prensky (2021) aponta que a aprendizagem baseada em jogos digitais reúne a diversão, do envolvimento em conjunto com a aprendizagem relacionando ao entretenimento interativo empolgante. Jogos que possuem essas características são chamados de jogos digitais para aprendizagem.

Os jogos PDG e UnH foram desenvolvidos dentro do projeto de pesquisa “Recurso Didático para Interação Humano-Computador”, são jogos digitais para aprendizagem com foco em conteúdos da disciplina IHC. Estes jogos são do tipo perguntas e respostas desenvolvidos para serem executados no navegador. Por serem jogos digitais, possuem interação constante com o usuário, e a usabilidade é um aspecto importante para esses sistemas. Neste trabalho, buscou-se dar continuidade a estes jogos, melhorando a sua usabilidade, ao adicionar alguns dos mecanismos de usabilidade.

Nas Subseções 2.3.1 e 2.3.2, são apresentadas informações sobre os jogos UnH e PDG.

2.3.1 Universidade das Heurísticas

Universidade das Heurísticas é um jogo digital para plataforma web, desenvolvido por Oliveira e Naves (2021). O jogo tem como objetivo apoiar o ensino e a aprendizagem de conceito de heurísticas de Nielsen.

O jogo é do tipo perguntas e respostas, *single player* (único jogador), em primeira pessoa, desenvolvido em 8-bits, no formato 2D. O público alvo são alunos de graduação e pós-graduação de cursos da área de Ciências da Computação.

Os estudantes devem responder perguntas de verdadeiro ou falso e de múltipla escolha, que ficam mais difíceis conforme o jogador avança de nível. Ao iniciar o jogo e chegar até a etapa de responder questões, a interface para responder questões será semelhante à Figura 4 do protótipo do jogo (captura de tela do protótipo do jogo).

Figura 4 – Universidade das Heurísticas.



Fonte: Oliveira e Naves (2021).

Ao se deparar com uma questão no jogo, é apresentada ao usuário uma interface similar a da Figura 4, onde é possível interagir respondendo a questão. Segundo BETHKE (2003), características como mecânica, fluxo ou regras são típicas do *Game Design Document* (GDD).

As especificações de projeto, sobre como implementar o jogo, abordando a arquitetura do *software*, modelagem de dados, requisitos, entre outros aspectos técnicos, segundo BETHKE (2003), estão presentes no *Technical Design Document* (TDD). A arquitetura do jogo é dividida em um *frontend* que utiliza o *framework* React ¹, e *backend* que utiliza as tecnologias Node.js, Express.js e o PostgreSQL com Banco de dados.

¹ React - Biblioteca para criação de interfaces de usuários, versão utilizada no projeto: 17.0.2. <https://pt-br.reactjs.org>

2.3.2 PersonaDesignGame

PersonaDesignGame é um jogo digital para plataforma web, desenvolvido por Junior (2021). O jogo tem como objetivo apoiar o processo de ensino e aprendizagem dos conceitos sobre personas da área de IHC, trazendo a prática do conteúdo através de perguntas e respostas.

O PDG é um jogo do tipo *single player* (único jogador), em primeira pessoa, no formato 2D, onde o público alvo são os estudantes da disciplina de IHC.

No GDD desenvolvido, para jogar, o estudante seleciona a opção de jogar (tela inicial Figura 5), então são apresentadas as fases disponíveis para o jogador. Clicando em uma fase, sendo essa desbloqueada, questões são apresentadas as quais devem ser respondidas em até cinco minutos.

Figura 5 – PersonaDesignGame.



Fonte: Junior (2021).

As perguntas podem variar entre múltiplas escolhas e verdadeiro ou falso. Para incentivo dos usuários, o jogo possui um ambiente de recompensas (prêmios).

Além de jogar respondendo as questões, o usuário também possui outras opções, apresentadas na Figura 5, como: conteúdo, onde é apresentado um resumo sobre o conceito de heurísticas, e prêmios, onde são apresentadas as premiações conquistadas pelo usuário ao concluir as fases.

O TDD apresentado por Junior (2021) detalha a arquitetura do jogo como sendo composta por microsserviços para o *backend*, contendo os serviços “persona-game-api” e “games-q&a-api”, desenvolvidos utilizando a tecnologia NodeJS e MySQL para banco de dados. No *frontend*, é utilizado o *framework* React.

2.4 Trabalhos Correlatos

Nesta seção são apresentados alguns trabalhos com propostas similares, encontrados na literatura, visando a melhoria na usabilidade de diferentes aplicações.

A pesquisa realizada por [Rodriguez, Acuña e Juristo \(2015\)](#) aborda questões de usabilidade com impacto em funcionalidades do sistema no processo de desenvolvimento. Três aplicações web foram desenvolvidas em diferentes linguagens, PHP, VB.NET e Java, com o intuito de descobrir padrões que implementam mecanismos de usabilidade. As soluções reutilizáveis definidas implementam dois mecanismos de usabilidade, sendo: abortar operação e *feedback* de operação.

Já no estudo realizado por [Luna et al. \(2010\)](#), os requisitos de usabilidade são abordados na engenharia da Web orientada a testes e baseada em modelos. O foco principal do estudo está nos requisitos de usabilidade com implicações funcionais, que podem ter impacto no *design* da arquitetura. No início de cada ciclo, é criado um conjunto de testes, contemplando os requisitos de usabilidade que orientam o desenvolvimento, garantindo que nenhuma funcionalidade relacionada seja alterada ao longo do ciclo de desenvolvimento.

Finalmente, [Capilla et al. \(2020\)](#) apresentam um estudo de caso em que analisam o impacto na usabilidade, de um aplicativo *mobile*, com a introdução de vários mecanismos de usabilidade. Para acomodar as mudanças requeridas pela implantação dos mecanismos de usabilidade, foi realizadas e relatadas mudanças arquiteturais no software. Após as mudanças da melhoria da usabilidade, foi realizada uma pesquisa de satisfação com diversos usuários para verificar se a aplicação dos mecanismos de usabilidade foi benéfica para usabilidade geral do sistema. Os esforços realizados para melhorias da usabilidade, no aplicativo móvel, mostraram-se efetivos ao final do estudo.

A partir dos trabalhos pesquisados é possível identificar alguns pontos positivos, como a utilização dos mecanismos de usabilidade para melhorias na usabilidade em diferentes aplicações e avaliações durante o ciclo de desenvolvimento do software. Diferentemente dos trabalhos relatados, este trabalho realizou avaliações e melhorias da usabilidade, aplicando mecanismos de usabilidade, em jogos educacionais que já concluíram a etapa de desenvolvimento do software.

2.5 Resumo do Capítulo

Os assuntos abordados no capítulo são a base para o projeto. Os conceitos, apresentados na Seção 2.1 (IHC), buscaram discutir assuntos como interação, interface e comunicabilidade, os quais são determinantes para a qualidade em uso de sistemas. A usabilidade é o principal foco da pesquisa, por isso, foi apresentada a sua decomposição

em características e seus subtipos (Seção 2.2).

A Seção 2.4 apresentou alguns trabalhos que têm como objetivo a melhoria da usabilidade em aplicações, utilizando alguns dos mecanismos de usabilidade apresentados na Seção 2.2. Estes trabalhos foram utilizados como apoio para a pesquisa, porém, diferentemente, esta pesquisa visa aplicar a melhoria na usabilidade ao fim do processo de desenvolvimento. Neste sentido, buscou-se melhorar a usabilidade dos jogos descritos nas Seções 2.3.1 e 2.3.2.

3 Metodologia

Este capítulo descreve a metodologia empregada para o desenvolvimento deste trabalho. A Metodologia apresentada a classificação da pesquisa realizada, fases e passos metodológicos utilizados neste projeto para alcançar os objetivos propostos.

Segundo [Filho e Filho \(2015\)](#), a classificação tem como objetivo auxiliar na compreensão do que se deve fazer, interpretar, e de que forma conduzir uma pesquisa. Com base na classificação proposta por [Filho e Filho \(2015\)](#), esta pesquisa possui campo **monodisciplinar** com foco na disciplina de IHC, especificamente na usabilidade, na qual os resultados são voltados para a **aplicação prática**, expressando o momento da realização da pesquisa (**estudo transversal**).

A pesquisa inicialmente é **exploratória**, onde o levantamento bibliográfico foi realizado, sendo complementada como pesquisa **descritiva** pela realização da observação sistemática dos jogos.

Os dados obtidos são representados numericamente, como número de mecanismos de usabilidades implementados nos jogos, caracterizando-se um tipo de dado **quantitativo**. Além disso, outros dados não possuem uma representação numérica, podendo ser classificados em categorias de acordo com os padrões analisados. A abordagem da pesquisa possui o tipo misto, tanto **quantitativa** quanto **qualitativa**.

O **estudo de caso** é o procedimento técnico utilizado, tendo o objetivo de avaliar a usabilidade dos jogos apresentados nas Seções [2.3.1](#) e [2.3.2](#). Os dados coletados são de caráter **primário**, sendo gerados a partir das avaliações realizadas sobre as aplicações.

A pesquisa foi dividida em quatro fases, conforme ilustrado na figura [6](#).

Figura 6 – Fases da pesquisa.



Fonte: Autores.

A seguir estão listados os **passos metodológicos** para alcançar os objetivos desta pesquisa, que foram apresentadas na Figura [6](#), sendo os passos:

1. Fase: *Elaboração do projeto de pesquisa*

- a) Pesquisa bibliográfica.
- b) Definição das unidades para o estudo de caso (jogos).
- c) Elaboração das questões utilizadas na avaliação.
- d) Estudo documental dos jogos (TCC).

2. Fase: *Entendimento dos jogos*

- a) Estudo documental dos jogos (TCC).
- b) Instalação e execução dos jogos.

3. Fase: *Coleta e avaliação*

- a) Planejamento da avaliação.
- b) Levantamento dos casos de uso.
- c) Aplicação da avaliação heurística individual nas unidades de estudos.
- d) Revisão da avaliação heurística em duplas nas unidades de estudos.
- e) Revisão dos resultados em grupo.

4. Fase: *Melhorias*

- a) Selecionar mecanismos a serem implementados a partir do resultado final da avaliação.
- b) Proposição de melhorias na usabilidade das unidades.
- c) Implementar melhorias.

Com a conclusão das fases descritas nos passos metodológicos, é possível concluir os objetivos específicos da Seção 1.2. O objetivo OE01 é alcançado com a realização das fases 1 e 2. O objetivo OE02 com a conclusão da fase 3. O objetivo OE03 é atingido com a realização do item a) da fase 4. O objetivo OE04 é alcançado com a realização dos itens b) e c) da fase 4. Cada uma das fases da pesquisa são descritas em detalhes nas seções seguintes.

3.1 Elaboração do Projeto de Pesquisa

De acordo com Gil (1995), o estudo de caso não possui um roteiro pré-definido com delimitação estabelecidas, mas pode ser dividido em quatro partes, sendo elas: (1) delimitação da unidade-caso; (2) coleta de dados; (3) seleção e interpretação dos dados; e (4) elaboração do relatório.

Considerando as fases definidas por Gil (1995), primeiramente, foi definido o tema para o estudo de caso juntamente com a delimitação das unidades, em que será aplicado

o estudo de caso. As **unidades-caso** definidas foram os jogos PersonaDesignGame e Universidade das Heurísticas. Esses jogos são detalhados nas Seções 2.3.2 e 2.3.1. Como são estudados dois casos, trata-se, portanto de um estudo de caso de múltiplos casos.

Para ser possível analisar os casos, é benéfico realizar a revisão bibliográfica buscando fundamentação teórica sobre tema de estudo e sobre os trabalhos semelhantes. No Capítulo 2, são registradas informações relevantes sobre essa etapa. Após a revisão bibliográfica iniciou-se, um estudo dos jogos, cujo resultado é apresentado na Seção 3.2.

Na coleta de dados e avaliação, vários mecanismos podem ser empregados, sendo eles dos tipos quantitativos e qualitativos. O detalhamento da coleta e avaliação é apresentado na Seção 3.3.

Já a elaboração de relatório é apresentada nos Capítulos 4 e 5, onde são especificados todos os dados coletados, categorizados e interpretados.

3.2 Entendimento dos Jogos

Esta etapa foi dividida em três partes (Subseções 3.2.1, 3.2.2 e 3.3) com o intuito de melhor entender os jogos dos quais foram coletados e analisados os dados.

3.2.1 Estudo da Documentação dos Jogos

O objetivo desta fase é entender os jogos analisados, através da documentação gerada por Junior (2021) e Oliveira e Naves (2021). Através dos documentos, é possível obter uma visão geral sobre a estrutura do jogo, arquitetura, tecnologias utilizadas e o processo de desenvolvimento.

Após ter a visão geral sobre os projetos, um estudo sobre as tecnologias utilizadas é realizado com intuito de alcançar uma melhor familiarização com as mesmas. Nesta etapa, a realização de treinamentos da dupla nas tecnologias utilizadas é de suma importância, tanto para o melhor entendimento sobre os padrões impostos pelas tecnologias quanto sobre a aplicação e suas limitações.

3.2.2 Instalação e Execução dos Jogos

A execução da fase anterior culminou na instalação e na execução local dos projetos. Nesta etapa, são utilizados os manuais de instalações disponibilizados pelos *frameworks*, ferramentas e dos projetos (jogos). Após a instalação, é possível executar as aplicações e analisar as funcionalidades descritas nas documentações. Portanto, nesta etapa, é possível verificar os requisitos para a execução dos projetos.

3.3 Coleta e Avaliação

De acordo com [Barbosa e Junqueira \(2010\)](#), a identificação dos objetivos de coleta de dados é essencial para definir quais dados devem ser coletados e quais técnicas podem ser empregadas. O objetivo principal da coleta é verificar quais mecanismos de usabilidade estão implementados nos jogos.

O foco da avaliação é buscar problemas na usabilidade do sistema. Com a definição do objetivo de avaliação, um conjunto de questões pode ser elaborado, associado ao objetivo.

Segundo [Barbosa e Junqueira \(2010\)](#), a avaliação realizada ao final do processo de *design*, quando já existe uma solução, é caracterizada como **avaliação somativa**. Portanto, as soluções envolvidas (jogos) já possuem os seus escopos definidos e implementados.

[Barbosa e Junqueira \(2010\)](#) classificou os métodos de avaliação de IHC sendo: métodos de investigação, de observação de uso e de inspeção. Para a avaliação dos jogos, o método utilizado é a inspeção, possibilitando que os avaliadores coloquem-se no lugar do usuário, tentando prever problemas na interface e na interação, permitindo avaliar a conformidade com os padrões.

Para a avaliação, na coleta de dados, é necessária a atividade de preparação, na qual são identificados os principais casos de uso dos jogos, descrevendo a interação entre usuário e sistema. Cada caso de uso possui uma tabela com as informações apresentadas na Tabela 3.

Tabela 3 – Estrutura caso de uso

Identificador	Código identificador do caso de uso
Nome	Nome do caso de uso
Pré-condições	Pré-condição necessária para o caso de uso
Fluxo principal	
Usuário	Sistema
Ações do usuário	Ações do Sistema
Fluxo alternativo	
Usuário	Sistema
Ações do usuário	Ações do Sistema

Fonte: Adaptado de [TIBOCHA \(2014\)](#).

Para cada caso de uso identificado nas aplicações, uma tabela similar à Tabela 3 foi gerada, contendo informações para a identificação e descrição dos casos de uso. Cada caso de uso possui identificador, nome, pré-condições e fluxos principais e alternativos. Caso

não haja fluxo alternativo, esta informação será omitida da tabela. Essa especificação é utilizada na avaliação dos jogos.

Na avaliação dos jogos, é necessária a padronização dos critérios a serem avaliados em relação aos mecanismos de usabilidade. Foi definida uma estrutura padrão de apoio para avaliação, representada na Tabela 4, com o intuito de apoiar os avaliadores, contendo as seguintes informações.

Tabela 4 – Estrutura padrão para apoio da avaliação.

Identificação
Nome: Nome do mecanismo de usabilidade
Família: Família a qual o mecanismo de usabilidade pertence, indicadas na tabela
Contexto
Contexto ao qual o mecanismo de usabilidade deve ser utilizado
Questão(ões)
Questão(ões) sobre como identificar se os requisitos estão sendo utilizado

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Para cada mecanismo de usabilidade presente na Tabela 1, uma tabela de apoio foi especificada com as informações necessárias para avaliação de acordo com a estrutura padrão representada na Tabela 4, conforme está descrito no Apêndice A.

Com os casos de usos definidos, foram utilizados na inspeção os conjuntos de questões presentes nas tabelas de apoio geradas para cada mecanismo de usabilidade.

Os dados produzidos são classificados em categorias (dados nominais), sendo eles:

- **Não implementado:** O mecanismo de usabilidade não está implementado para a funcionalidade.
- **Parcialmente implementado:** O mecanismo de usabilidade não está implementado completamente para a funcionalidade.
- **Implementado:** O mecanismo de usabilidade está implementado completamente para a funcionalidade.
- **Não se aplica:** O mecanismo de usabilidade não se aplica para a funcionalidade.

Para a realização da avaliação heurística (inspeção), [Nielsen \(1992\)](#) recomenda que o número de avaliadores seja de 3 a 5. [Barbosa e Junqueira \(2010\)](#) apresentam, na inspeção heurísticas, tarefas que devem ser realizadas individualmente e outras em conjunto. Na avaliação realizada, buscou-se seguir as recomendações utilizando mais de três avaliadores em determinadas atividades, auxiliando na validação obtida pela dupla de avaliadores (autores).

Na inspeção, inicialmente, a dupla de avaliadores (autores), seleciona o caso de uso que será avaliado. Após esta etapa, o caso de uso é avaliado individualmente com base nas questões definidas para a inspeção. Com o resultado individual obtido, os avaliadores reunirão-se para realização da revisão dos resultados.

A Tabela 5 apresenta o modelo utilizado como apoio na inspeção:

Tabela 5 – Estrutura apoio inspeção.

Questão	Casos de Uso
	Identificador(es) caso(s) de uso
Questão a ser respondida	Classificação

Fonte: Autores.

A estrutura apresentada na Figura 5 é utilizada na avaliação do mecanismo de usabilidade, associando as questões, os casos de usos e sua classificação. As questões utilizadas estão presentes nas tabelas de apoio, no Apêndice A.

Para cada problema encontrado, uma descrição é realizada, identificando o caso de uso, local em que o problema se encontra, descrição dos problemas, sua classificação e imagens da interface que apresentam o problema. Para a identificação dos problemas encontrados, será utilizada a estrutura a seguir:

Tabela 6 – Estrutura padrão identificação de violação.

Caso de Uso	Identificador- Nome		
Local	Página(s) e/ou componente(s)		
Problema(s)	Descrição(ões)		
Classificação	Classificação	Figura(s)	Referência figura(s)

Fonte: Autores.

Concluindo as atividades para todos os casos de uso, outros avaliadores (externos), que trabalham com o mesmo tema, podem ser consultados para validar os resultados originados. A revisão em grupo, quando possível, é realizada até que se atinja um índice de consenso mínimo. Para isso, é utilizado o índice **Kappa de Cohen** determinando o grau de concordância entre os avaliadores. A fórmula para o calculo é representada a seguir.

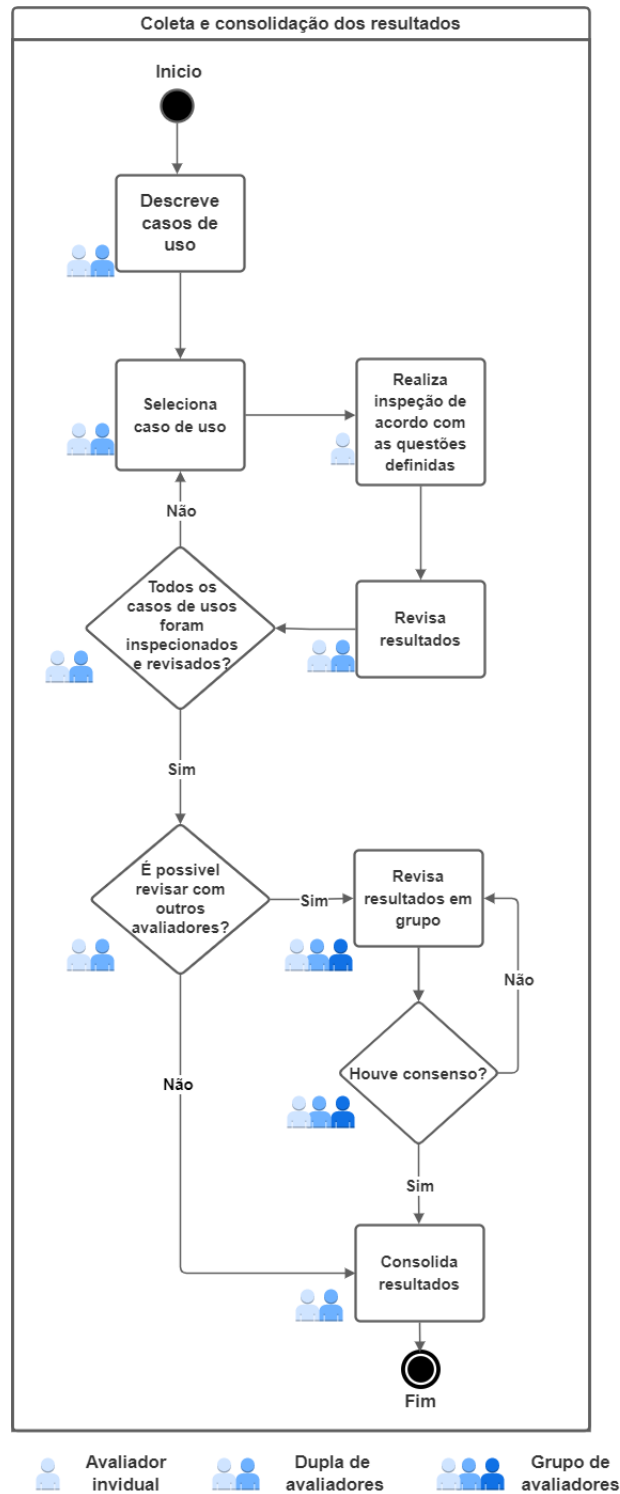
$$\kappa = (Po - P\epsilon)/(1 - P\epsilon) \quad (3.1)$$

Ao avaliar a concordância utilizando o Kappa, =foi obtido um índice entre 0 e 1, onde 1 significa que a concordância entre os avaliadores aproximou-se da perfeição. Para esta pesquisa, utilizou-se a interpretação da magnitude do coeficiente Kappa definida por Landis e Koch (1977), apresentada na Tabela 134 no Anexo A.

A revisão em grupo é considerada satisfeita quando a **concordância ultrapasse 0.6** do índice kappa, significando uma concordância substancial entre os avaliadores, de acordo com [Landis e Koch \(1977\)](#).

As atividades realizadas na coleta e avaliação são representadas no diagrama, ilustrado na [Figura 7](#).

Figura 7 – Coleta e consolidação dos resultados.



Fonte: Autores.

As atividades realizadas, apresentadas na Figura 7, envolvendo vários avaliadores, tem como principal objetivo mitigar falhas no processo de inspeção, trazendo maior confiabilidade aos resultados obtidos.

3.4 Melhorias

Nesta seção são apresentadas as estratégias para melhorias na usabilidade ao jogo.

A saída da etapa de coleta e Avaliação (Seção 3.3) é uma lista de problemas de usabilidade organizador por mecanismo de usabilidade. Com isso, é possível dar início a fase de melhoria, que tem como foco solucionar esses problemas.

Para cada um dos problemas, é proposta uma melhoria a fim de atender os requisitos de cada mecanismo de usabilidade desta família. A proposta de melhoria poderá ser realizada sobre interface, diagramações, prototipação ou arquitetura. As melhorias elaboradas estão presentes no Capítulo 6.

3.5 Resumo do Capítulo

Neste capítulo foi apresentada a classificação da pesquisa realizadas neste trabalho, bem como os métodos, processos e técnicas, onde foram apresentadas as fases e passos metodológicos seguidos no desenvolvimento do projeto.

A pesquisa foi definida em quatro fases sendo elas: Elaboração do projeto de pesquisa (Seção 3.1); Entendimento dos jogos (Seção 3.2); Coleta e avaliação (Seção 3.3) e Melhorias (Seção 3.4). Para cada uma das fases foi reservada uma seção especificando o processo e suas atividades de forma detalhada, seguidas posteriormente no desenvolver do projeto.

A metodologia apresentada neste capítulo, define o processo utilizado para a realização da melhora na usabilidade dos jogos. Portanto, especificando desde a identificação da necessidade de melhoria até a sua implementação.

4 Avaliação do jogo PDG

Neste capítulo, é apresentada a avaliação do jogo PDG, bem como os detalhes utilizados nas avaliações em dupla e em grupo sobre o jogo. Sendo assim, resultando na classificação dos mecanismos de usabilidade presentes no PDG.

4.1 Coleta de Dados

Neste seção são apresentadas os preparativos e detalhes das avaliações em dupla e em grupo. Desta forma, é obtido como resultado a classificação dos mecanismos de usabilidade no jogo PDG.

4.1.1 Preparação

Nesta subseção, é realizada a preparação para a coleta de dados, identificando os principais casos de uso presentes nos estudos de casos definidos, o jogo PersonaDesignGame.

A descrição do jogo PersonaDesignGame foi apresentada na Seção 2.3.2. Os casos de uso identificados no jogo são:

- **UC01:** Responder questões.
- **UC02:** Visualizar conteúdo.
- **UC03:** Visualizar recompensas.

As Tabelas 7, 8 e 9 descrevem em detalhes cada caso de uso identificado no jogo PDG

Tabela 7 – UC01-Responder questões

Identificador	UC01	
Nome	Responder questões	
Pré-condições	Ter usuário gerado pela aplicação	
Fluxo principal		
Usuário	Sistema	
1. Usuário clica no botão de jogar 3. Usuário seleciona uma etapa desbloqueada 5. Usuário seleciona a opção e clica em confirmar 7. Usuário clica em continuar	2. O sistema apresenta as fases desbloqueadas, bloqueadas e suas sub etapas 4. O sistema apresenta a questão a ser respondida 6. Sistema apresenta um <i>feedback</i> sobre a correção da questão 8. Sistema apresenta a próxima questão	
Fluxo alternativo		
Usuário	Sistema	
1. Usuário clica no botão de jogar 3. Usuário seleciona uma etapa desbloqueada 5. Usuário seleciona a opção e clicando em confirmar 7. Usuário clica no botão anterior para voltar a questão anterior	2. O sistema apresenta as fases desbloqueada, bloqueadas e suas etapas 4. O sistema apresenta a questão a ser respondida 6. Sistema apresenta um <i>feedback</i> sobre a correção da questão 8. O sistema apresenta a questão anterior	

Fonte: Autores.

Tabela 8 – UC02-Visualizar conteúdo

Identificador	UC02	
Nome	Visualizar conteúdo	
Pré-condições	Ter usuário gerado pela aplicação	
Fluxo principal		
Usuário		Sistema
1. Usuário clica no botão conteúdo		2. Sistema apresenta resumo do conteúdo sobre o assunto
Fluxo alternativo		
Usuário		Sistema
1. Usuário clica no botão de jogar 3. Usuário seleciona uma etapa desbloqueada 5. Usuário seleciona a opção e clica em confirmar 7. Usuário clica no botão anterior para voltar a questão anterior		2. O sistema apresenta as fases desbloqueada e bloqueadas e suas etapas 4. O sistema apresenta a questão a ser respondida 6. O Sistema apresenta um <i>feedback</i> sobre a correção da questão 8. O sistema apresenta a questão anterior

Fonte: Autores.

Tabela 9 – UC03-Visualizar recompensas

Identificador	UC03	
Nome	Visualizar recompensas	
Pré-condições	Ter usuário gerado pela aplicação Possuir fase desbloqueada	
Fluxo principal		
Usuário		Sistema
1. Usuário clica em prêmios 3. O usuário seleciona a opção de medalhas		1. Sistema apresenta as recompensas e cartões de persona 4. O sistema apresenta as medalhas

Fonte: Autores.

Nas Tabelas 7, 8 e 9, são especificados os principais casos de uso do jogo PDG, utilizados na avaliação.

4.2 Avaliação

Para a coleta dos dados, cada avaliador realizou a inspeção das interfaces apresentadas, compreendendo os passos presentes nos casos de uso. As avaliações individuais estão presentes em: [PDG - Avaliações individuais](#).

Com o resultado individual dos avaliadores, realizou-se a revisão em dupla, respeitando as atividades apresentadas na avaliação, como sugerido por (BARBOSA; JUN-

QUEIRA, 2010). Após a revisão em dupla, foi realizada a consolidação dos resultados em grupo.

4.2.1 Revisão dos Resultados: Dupla (PDG)

Nesta subseção estão apresentados os resultados obtidos após a revisão em dupla das avaliações individuais. As Tabelas de 38 até 62 (Apêndice B.1) contém a revisão em dupla com as respostas e a identificação dos problemas encontrados para cada mecanismo avaliado, resultando na classificação da implementação dos mecanismos no PDG:

Tabela 10 – Revisão em dupla-PDG

Família	Mecanimo	Classificação
Feedback	Interação	Parcialmente implementado
	<i>Feedback</i> para ação longa	Não se aplica
	Aviso	Não implementado
	Estado do sistema	Parcialmente implementado
Cancelar/desfazer	Desfazer global	Não se aplica
	Desfazer em objeto específico	Não implementado
	Abortar Operação	Parcialmente implementado
	Voltar	Parcialmente implementado
Prevenção/correção de erros de entrada de usuário	Entrada de texto estruturado	Não se aplica
Assistente	Execução Passo-a-Passo	Não se aplica
Perfil do usuário	Preferências	Não Implementado
	Espaço de objeto pessoal	Não Implementado
	Favoritos	Não Implementado
Ajuda	Ajuda multinível	Não se aplica
Agregação de comando	Agregação de comando	Não se aplica

Fonte: Autores.

Os dados apresentados na Tabela 10, com os resultados da revisão em dupla do jogo PDG, apontam que a dupla classificou a maioria dos mecanismos (9 de 15) com alguma deficiência ou necessidade de implementação.

4.2.2 Revisão dos Resultados: Grupo (PDG)

A **primeira revisão em grupo** foi realizada pela dupla 1 (autores) e dupla 2 (Ailamar e Heron)¹, onde as duplas trabalharam na avaliação da usabilidade no jogo PDG simultaneamente. As duplas reuniram-se para revisar os resultados obtidos pela avaliação gerada por cada uma das duplas.

Na Tabela 11, são apresentados os resultados da primeira revisão em grupo, onde estão descritas as concordâncias e discordâncias entre as duplas.

Tabela 11 – Primeira revisão em grupo

Mecanismo	Dupla 1	Dupla 2	Revisão
Interação	Parcialmente implementado	Parcialmente implementado	Concordância: Existe a falta de comunicação em algumas ações do jogo para informar ao usuário. A interação existe apenas na resposta da partida em caso de acerto ou erro e quando usuário passa de fase. O sistema fornece uma ação visual em alguns eventos com o mouse, porém não contém um <i>feedback</i> explicativo da ação, como mudança visual no componente ou texto explicativo.
Desfazer global	Não se aplica	Não se aplica	Concordância: O jogo não possui tarefas complexas onde uma etapa deve ser desfeita/revertida.
Desfazer específico	Não implementado	Não se aplica	Discordância: A dupla 1 não identificou que era possível desfazer uma tarefa, já a dupla 2 identificou que não era possível.
Abortar operação	Parcialmente implementado	Implementado	Discordância: Algumas questões utilizadas na avaliação são diferentes entre as equipes.

¹ Dupla 2 - Ailamar Alves Guimarães e Heron Rodrigues Sousa, estudantes do curso de Engenharia de Software na UnB, desenvolveram um trabalho na área de IHC com o jogo PDG. Estes estudantes foram convidados para a avaliação do jogo PDG pela experiência e tema de trabalho.

Voltar	Parcialmente implementado	Implementado	Discordância: Cada dupla usou questões com foco em pontos diferentes. Voltar a um estado de segurança e outra voltar a uma espaço (interface) específica.
Estado do sistema	Parcialmente implementado	Não implementado	Discordância: Dupla 2 acredita que o estado do sistema não foi informado, já a dupla 1 acredita que há um <i>feedback</i> discreto no sistema através do <i>loading</i> .
Aviso	Não implementado	Não implementado	Concordância: Não é informado ao usuário a consequência das ações ao realizar uma tarefa e sua confirmação.
Informar sobre longa ação	Não se aplica	Não se aplica	Concordância: A simplicidade do sistema permite que o jogo rode sem ser necessário esse tipo de funcionalidade. Não possui tarefas com tempo de espera maior que 2 segundos.
Entrada de texto estruturada	Não se aplica	Não se aplica	Concordância: A interação com a aplicação somente ocorre através de cliques com o mouse, não utiliza entradas de texto.
Execução passo a passo	Não se aplica	Parcialmente implementado	Discordância: Dupla 1 reconhece que o jogo não possui tarefa complexa pouco frequente. A dupla 2 levou em consideração tarefas que possuem sub-tarefas, ao invés de sua complexidade.
Preferências	Não implementado	Não implementado	Concordância: Não é possível determinar e gravar as preferências do usuário no jogo.

Áreas de objetos pessoais	Não implementado	Não se aplica	Discordância: Dupla 1 considera que a aplicação não realiza a coleta de dados do usuário, para criar esse espaço. A dupla 2 considera que não aplica pela simplicidade do jogo.
Favoritos	Não implementado	Não se aplica	Discordância: Dupla 1 considera que a aplicação não realiza a coleta de dados do usuário, para criar esse espaço. Dupla 2 considera que não aplica pela simplicidade do jogo.
Ajuda multi-nível	Não implementado	Não implementado	Concordância: O jogo não fornece ajuda ao realizar qualquer tarefa.
Agregação de comando	Não se aplica	Não se aplica	Concordância: Não possui ações longa sequência de ações em que o usuário precisa repeti-las

Fonte: Autores.

Na primeira revisão em grupo, as duplas chegaram em acordo em relação a alguns mecanismos, porém em outros não. Na maioria dos casos, não foi possível encontrar um acordo, por conta das duplas possuírem questões diferentes para a avaliação.

Após a verificação dos resultados obtidos pelas duplas (Tabela 11) na primeira revisão, verificou-se o grau de consenso obtido. A Tabela 12 apresenta, em números, os dados utilizados para o cálculo do índice kappa.

Tabela 12 – Consenso primeira revisão

Dupla 2	Dupla 1				Total
	Implementado	Parcialmente implementado	Não implementado	Não aplica	
Implementado	0	0	0	0	0
Parcialmente implementado	2	1	1	0	4
Não implementado	0	0	2	3	5
Não aplica	0	1	0	4	5
Total	2	2	3	7	14

Fonte: Autores.

Com os dados apresentados na Tabela 12, é realizado o cálculo de consenso entre as duplas.

$$\kappa = (Po - P\epsilon)/(1 - P\epsilon) \quad (4.1)$$

$$Po = (0 + 1 + 2 + 4)/14 = 0,50 \quad (4.2)$$

$$P\epsilon = (2/14) * (0/14) + (2/14) * (4/14) + (3/14) * (5/14) + (5/14) * (7/14) \approx 0,29 \quad (4.3)$$

$$\kappa = (0,50 - 0,29)/(1 - 0,29) \approx 0,29 \quad (4.4)$$

O resultado obtido demonstra que o consenso entre as equipes está razoável, entre 0,20 a 0,40. Porém, para dar continuidade, espera-se que o grau de concordância esteja próximo da concordância substancial (0,60) ou perfeita (1,00). Para isso, foi revisada a avaliação em grupo para cada mecanismo, visando um aumento significativo no consenso.

A **segunda revisão em grupo** visou os pontos onde as duplas tiveram discordâncias, apontadas na primeira revisão (Tabela 11). Os resultados obtidos com a segunda revisão estão apresentados na Tabela 13.

Tabela 13 – Segunda revisão em grupo

Mecanismo	Dupla 1	Dupla 2	Revisão
Desfazer específico	Não se aplica	Parcialmente implementado	Consenso: Não conseguimos avaliar pois não temos informações suficientes na documentação nem interface para julgar.
Abortar operação	Parcialmente implementado	Implementado	Não foi possível chegar a um consenso entre as duplas.
Voltar	implementado	implementado	Consenso: O jogo possibilita voltar mas não informa o estado (outro mecanismo).
Estado do sistema	Parcialmente implementado	Parcialmente implementado	Consenso: O jogo possibilita voltar mas não informa o estado (outro mecanismo).
Execução passo a passo	Não se aplica	Parcialmente implementado	Não foi possível chegar a um consenso entre as duplas.

Tabela 13 – Segunda revisão em grupo

Mecanismo	Dupla 1	Dupla 2	Revisão
Áreas de objetos pessoais	Não se aplica	Não se aplica	Consenso: De acordo com a simplicidade do jogo não é necessária essa funcionalidade.
Favoritos	Não se aplica	Não se aplica	Consenso: De acordo com a simplicidade do jogo não é necessária essa funcionalidade.

Fonte: Autores.

A Tabela 13 apresentou os resultados da segunda revisão, onde é possível verificar que as duplas entraram em consenso na maioria das discordâncias identificadas na Tabela 11. Porém, nos mecanismos desfazer específico, abortar operação e execução passo-a-passo, o consenso não foi encontrado, permanecendo o julgamento da dupla (Autores).

Após a verificação dos resultados obtidos pelas duplas (Tabela 13) na segunda revisão, verificou-se o grau de consenso obtido. A Tabela 14 apresenta, em números, os dados utilizados para o cálculo do índice kappa.

Tabela 14 – Consenso segunda revisão

Dupla 2	Dupla 1				Total
	Implementado	Parcialmente implementado	Não implementado	Não aplica	
Implementado	1	0	0	0	1
Parcialmente implementado	1	1	0	0	2
Não implementado	0	0	4	0	4
Não aplica	0	0	0	7	7
Total	2	1	4	7	14

Fonte: Autores.

Com os dados apresentados na Tabela 14, foi realizado o cálculo de consenso entre as duplas.

$$\kappa = (Po - P\epsilon)/(1 - P\epsilon) \quad (4.5)$$

$$Po = (1 + 1 + 4 + 7)/14 \approx 0.93 \quad (4.6)$$

$$P\epsilon = (2/14) * (1/14) + (1/14) * (2/14) + (4/14) * (4/14) + (7/14) * (7/14) \approx 0.35 \quad (4.7)$$

$$\kappa = (0,93 - 0,35)/(1 - 0,35) \approx 0,89 \quad (4.8)$$

Na segunda revisão, após discussão sobre os pontos de discordâncias encontrados, entrou-se em consenso sobre a avaliação de vários mecanismos. Com isso, obteve-se um índice de consenso de 0,89. O consenso obtido é classificado como consenso quase perfeito, deste modo, superando a meta prevista.

4.2.2.1 Dificuldade Revisão em Grupo

Na avaliação as duplas, em alguns casos, utilizaram questões diferentes por conta de diferentes interpretações das referências utilizadas. Sendo assim, houve dificuldade para encontrar consenso entre as duplas. Em poucos casos, o consenso não foi atingido, prevalecendo a interpretação original da dupla.

4.2.2.2 Resultado Revisão em Grupo

A revisão em grupo foi importante para identificar possíveis incoerências na revisão individual e em dupla. Na primeira revisão, foram reconhecidas algumas divergências nas avaliações de cada dupla. Na segunda revisão, as duplas entraram em consenso sobre várias divergências, sendo o principal motivo interpretações diferentes dos mecanismos de usabilidade. O resultado da revisão em grupo está apresentada na Tabela 15.

Tabela 15 – Resultado revisão em grupo-PDG

Família	Mecanimo	Classificação
Feedback	Interação	Parcialmente implementado
	<i>Feedback</i> para ação longa	Não se aplica
	Aviso	Não implementado
	Estado do sistema	Parcialmente implementado
Cancelar/desfazer	Desfazer global	Não implementado
	Desfazer em objeto específico	Não se aplica
	Abortar Operação	Parcialmente implementado
	Voltar	Implementado
Prevenção/correção de erros de entrada de usuário	Entrada de texto estruturado	Não se aplica
Assistente	Execução Passo-a-Passo	Não se aplica
Perfil do usuário	Preferências	Não Implementado
	Espaço de objeto pessoal	Não se aplica
	Favoritos	Não se aplica
Ajuda	Ajuda multinível	Não se aplica
Agregação de comando	Agregação de comando	Não se aplica

Fonte: Autores.

Os problemas relatados pelas duplas permaneceram similares, como apresentado na Tabela 15, sendo assim os problemas identificados pela dupla foram mantidos. Os Capítulos 6 e 7 apresentarão as melhorias para os mecanismos de usabilidades escolhidos, não implementados totalmente.

4.3 Resumo do Capítulo

Neste capítulo foram apresentados os dados colhidos através da avaliação do jogo PDG, juntamente com as revisões em dupla e em grupo. Deste modo, trazendo a visão de dois especialistas externos à pesquisa, mas que estão trabalhando sobre o mesmo assunto. Portanto, a revisão dos resultados em grupo trouxe maior confiabilidade na avaliação realizada.

5 Avaliação do jogo UnH

Neste capítulo, é apresentada a avaliação do jogo UnH, bem como os detalhes utilizados nas avaliações individuais e em dupla, resultando na classificação dos mecanismos de usabilidade presentes no UnH.

5.1 Coleta de Dados

Neste seção são apresentados os preparativos e detalhes das avaliações individuais e em dupla. Desta forma, é obtido como resultado a classificação dos mecanismos de usabilidade no jogo UnH.

5.1.1 Preparação

A descrição do jogo UnH foi apresentada na Seção 2.3.1. Os casos de uso identificados no jogo estão listados a seguir:

- **UC01:** Cadastrar usuário.
- **UC02:** Responder questões.
- **UC03:** Visualizar *ranking*.
- **UC04:** Informações sobre o jogo.

As Tabelas 16, 17, 18 e 19 descrevem em detalhes cada caso de uso identificados no jogo UnH, para orientar na avaliação das interfaces.

Tabela 16 – UC01- Cadastrar usuário

Identificador	UC01	
Nome	Cadastrar usuário	
Pré-condições	Nenhuma	
Fluxo principal		
Usuário	Sistema	
1. Usuário seleciona opção começar 2. Usuário seleciona opção jogar 4. Usuário insere o nome 5. Usuário seleciona personagem 6. Usuário confirma criação de usuário	3. O sistema solicita informações do usuário 7. O sistema cria usuário	

Fonte: Autores.

Tabela 17 – UC02- Responder questões

Identificador	UC02	
Nome	Responder questões	
Pré-condições	Possuir usuário criado	
Fluxo principal		
Usuário	Sistema	
1. Usuário seleciona fase desbloqueada 3. Usuário confirma as instruções 5. Usuário responde a questão 7. Usuário prossegue para próxima questão	2. O sistema apresenta instruções da fase 4. O sistema apresenta uma questão 6. O sistema apresenta feedback da resposta (em caso de erro a resposta correta e acerto a pontuação) 8. O sistema apresenta próxima questão	

Fonte: Autores.

Tabela 18 – UC03- Verificar *ranking*

Identificador	UC03	
Nome	Verificar ranking	
Pré-condições	Usuário estar respondendo pergunta	
Fluxo principal		
Usuário	Sistema	
1. Usuário seleciona a opção começar 3. Usuário seleciona a opção <i>ranking</i>	2. Sistema apresenta as opções do jogo 4. Sistema apresenta o <i>ranking</i> do jogo	
Fluxo alternativo		
Usuário	Sistema	
1. Usuário inicia uma fase 3. Usuário seleciona opção <i>ranking</i>	2. Sistema exibe pergunta 4. Sistema apresenta o <i>ranking</i> do jogo	

Fonte: Autores.

Tabela 19 – UC04- Informações sobre o jogo

Identificador	UC04	
Nome	Informações sobre o jogo	
Pré-condições	Nenhuma	
Fluxo principal		
Usuário	Sistema	
1. Usuário seleciona opção começar 3. Usuário seleciona opção sobre nós	2. Sistema apresenta as opções do jogo 4. sistema exibe informações sobre o jogo	

Fonte: Autores.

Os casos de usos apresentados nas Tabelas 16, 17, 18 e 19 são utilizados na avaliação no UnH, através dos fluxos definidos, é possível realizar a avaliação das interfaces.

5.2 Avaliação

Para a coleta de dados, cada avaliador realizou a inspeção das interfaces, cumprindo o passo a passo descrito nos casos de uso. Diferentemente da avaliação do PDG, a avaliação contou somente com a avaliação individual e em dupla, em virtude da falta de outros especialistas para compor a avaliação. Entretanto, sem comprometer os resultados, visto a experiência adquirida pelos avaliadores durante a avaliação do outro jogo. As avaliações realizadas estão presentes no Apêndice C.

5.2.1 Revisão dos Resultados: Dupla (UnH)

Nesta subseção estão apresentados os resultados obtidos após a revisão em dupla das avaliações individuais. No Apêndice C.3, contém a revisão em dupla com as respostas e identificações dos problemas encontrados para cada mecanismo avaliado. A Tabela 20 contém informações da revisão realizada pela dupla.

Tabela 20 – Primeira revisão em dupla

Mecanismo	João	Moacir	Revisão
Interação	Parcialmente implementado	Parcialmente implementado	Alguns botões mudam de opacidade ao passar o mouse, porém não possuem confirmação de interação aceita. Alguns botões e cartões, quando estão em foco, não apresentam mudanças visuais. Deste modo, o mecanismo está implementado parcialmente.
Desfazer global	Não Implementado	Não se aplica	O jogo não possui tarefas de múltiplas etapas que precisam ser desfeitas. Deste modo, não se aplica para este mecanismo.

Tabela 20 – Primeira revisão em dupla

Mecanismo	João	Moacir	Revisão
Desfazer específico	Não Implementado	Não se aplica	O jogo não possui uma tarefa a ser desfeita ou revertida, por se tratar de um jogo simples em que a funcionalidade principal responder questões não tem o intuito de ser desfeita. Deste modo, não se aplica para este mecanismo.
Abortar operação	Parcialmente implementado	Não implementado	Não é possível abordar a operação de criar usuário. Esta operação possui a possibilidade de cancelamento das questões, porém não a confirmação do cancelamento. Portanto, o mecanismo é parcialmente implementado.
Voltar	Parcialmente implementado	Não se aplica	Parcialmente implementado, pois o jogo possibilita voltar a estados anteriores como a tela de projetos, mas no cadastro de usuário não é possível a realização desta ação.
Estado do sistema	Parcialmente implementado	Não implementado	Ao realizar uma interação, não é informado que a informação está sendo processada, nem mesmo apresentação de uma falha no sistema. Sendo assim, o mecanismo está não implementado.
Aviso	Não implementado	Não implementado	Ações onde os dados seriam gravados ou alterados e tarefas que demandam muito tempo ao usuário não informam as consequências das ações. O usuário não é informado sobre o grau de consequência das ações. Portanto, o mecanismo não está implementado.

Tabela 20 – Primeira revisão em dupla

Mecanismo	João	Moacir	Revisão
Informar sobre longa ação	Não se aplica	Não se aplica	O sistema não possui ações com tempo de processamento maior de 2 segundos. Desta forma, o mecanismo não se aplica.
Entrada de texto estruturada	Parcialmente implementado	Parcialmente implementado	O número de cadastro é limitado, mas a descrição não informa. Também permite caracteres como espaços. Portanto, o mecanismo não está implementado.
Execução passo a passo	Não se aplica	Parcialmente implementado	O jogo indica informações incompletas sobre as etapas a serem realizadas. Deste modo, o mecanismo está parcialmente implementado.
Preferências	Não implementado	Não se aplica	A aplicação não coleta informações suficientes do usuário para este mecanismo. Não possui um perfil reutilizável. Portanto, este mecanismo não se aplica.
Espaço de objetos pessoais	Não implementado	Não se aplica	A aplicação não coleta informações suficientes do usuário para este mecanismo, não possuindo um perfil reutilizável e não é algo idealizado para o jogo com simples funcionalidades. Sendo assim, o mecanismo não se aplica.
Favoritos	Não se aplica	Não se aplica	A aplicação não coleta informações suficientes do usuário para este mecanismo. Não possui um perfil reutilizável, pois não foi uma funcionalidade idealizada para o jogo, com simples funcionalidades. Desta forma, não se aplica para estes mecanismos.

Tabela 20 – Primeira revisão em dupla

Mecanismo	João	Moacir	Revisão
Ajuda multi-nível	Não implementado	Parcialmente implementado	A informação de como realizar as tarefas estão incompletas. Portanto, está parcialmente implementado.
Agregação de comando	Não se aplica	Não se aplica	Por se tratar de um jogo simples de perguntas e respostas, não possui sequências de ações longas que seja possível a agregação de comandos. Desta forma, não se aplica para este mecanismo

Fonte: Autores.

Após a verificação dos resultados obtidos individualmente (Tabela C.3), foi verificado o grau de consenso obtido. A Tabela 21 apresenta, em números, os dados utilizados para o cálculo do índice kappa.

Tabela 21 – Consenso primeira revisão

João	Moacir				Total
	Implementado	Parcialmente implementado	Não implementado	Não aplica	
Implementado	0	0	0	0	0
Parcialmente implementado	0	3	0	0	3
Não implementado	0	0	4	0	4
Não aplica	0	0	4	4	8
Total	0	3	8	4	15

Fonte: Autores.

Com os dados apresentados na Tabela 21, foi realizado o cálculo de consenso entre os membros:

$$\kappa = (P_o - P_e)/(1 - P_e) \quad (5.1)$$

$$P_o = (0 + 3 + 4 + 4)/15 \approx 0.73 \quad (5.2)$$

$$P_e = (0/15) * (0/15) + (3/15) * (3/15) + (8/15) * (4/15) + (4/15) * (8/15) \approx 0.32 \quad (5.3)$$

$$\kappa = (0,73 - 0,32)/(1 - 0,32) \approx 0,60 \quad (5.4)$$

Na primeira revisão, o coeficiente de consenso Kappa foi aproximadamente 0,60. O consenso obtido é classificado como consenso moderado, de acordo com a Tabela 134. Desta forma, atingindo o grau de concordância esperado pela dupla.

5.2.1.1 Resultado Revisão em Dupla

A revisão em dupla foi importante para identificar possíveis incoerências na revisão individual. Na primeira revisão, foram reconhecidas algumas divergências nas avaliações individuais. O resultado da avaliação em grupo está apresentado na Tabela 22.

Tabela 22 – Revisão em dupla - UnH

Família	Mecanismo	Classificação
Feedback	Interação	Parcialmente implementado
	<i>Feedback</i> para ação longa	Não se aplica
	Aviso	Não implementado
	Estado do sistema	Não implementado
Cancelar/desfazer	Desfazer global	Não se aplica
	Desfazer em objeto específico	Não se aplica
	Abortar Operação	Parcialmente implementado
	Voltar	Não se aplica
Prevenção/correção de erros de entrada de usuário	Entrada de texto estruturado	Não implementado
Assistente	Execução Passo-a-Passo	Implementado Parcialmente
Perfil do usuário	Preferências	Não se aplica
	Espaço de objeto pessoal	Não se aplica
	Favoritos	Não se aplica
Ajuda	Ajuda multinível	Parcialmente implementado
Agregação de comando	Agregação de comando	Não se aplica

Fonte: Autores.

A classificação final da dupla é apresentada na Tabela 22, onde a dupla chegou em consenso entre todos os mecanismos. Na próxima seção, será apresentado as melhorias para os mecanismos de usabilidades escolhidos.

5.3 Resumo do Capítulo

Neste capítulo foram apresentados os dados colhidos através da avaliação do jogo do UnH, juntamente com as revisões em dupla. A avaliação possibilitou determinar o

grau de implementação dos mecanismos no jogo. Portanto, são possíveis a seleção e o planejamento das melhorias para os mecanismos selecionados.

6 Planejamento das Melhorias nos Jogos

Neste capítulo são apresentados os mecanismos selecionados para melhorias, as melhorias, bem como foram arquitetadas a fim de implementar os mecanismos nos Jogos UnH e PDG.

Com os resultados das avaliações dos jogos PDG e UnH, os autores selecionaram mecanismos classificados como não implementado ou parcialmente implementado para propor melhorias.

6.1 Mecanismos Escolhidos

As avaliações indicaram que os jogos possuem deficiências em diversos mecanismos, apresentando problemas semelhantes em alguns mecanismos. Sendo assim, os mecanismos selecionados pelos autores para melhorias nos dois jogos foram:

- **Interação**
- **Aviso**
- **Estado do sistema**

Além dos mecanismos indicados anteriormente, para o jogo UnH também foram incluídos os mecanismos:

- **Entrada de texto estruturado**
- **Execução Passo-a-Passo**

Estes mecanismos estão parcialmente implementados, onde correções pontuais podem ser feitas para que o mecanismo seja implementado.

6.2 Arquitetando Melhorias

Os problemas encontrados nos mecanismos de usabilidade selecionados ocorrem, em sua maioria, pela falta de apresentação de informações importantes ao usuário, sejam nos componentes botões, modais ou caixas de textos. A seguir, estão apresentadas as propostas de melhorias para a implementação total dos mecanismos selecionados para os jogos PDG e UnH.

6.2.1 Interação

Os problemas identificados nas Tabelas 41 e 114 apresentam as necessidades de melhorias das interações em botões. As melhorias para este componente foram:

- **Alteração de cores e opacidade:** Oferece maior destaque ao botão que está em alvo.
- **Texto informativo:** Descreve de maneira breve a ação a ser realizada.
- **Aviso:** Informa ao usuário que a interação gerou um resultado.

6.2.2 Aviso

Os problemas identificados nas Tabelas 44 e 118 apresentam a necessidade de avisos ao usuário ao realizar ações que gerem consequência ao mesmo, como a perda de dados. Os principais problemas foram encontrados na interação com botões, sendo propostas as melhorias para estes componentes:

- **Informações de consequências:** Apresenta informações sobre as consequências geradas ao realizar a ação.
- **Confirmação da ação:** É solicitado ao usuário se pretende continuar a ação, mesmo sendo informado sobre as consequências.

6.2.3 Estado dos Sistema

Os principais problemas identificados nas Tabelas 39 e 113 apresentam a necessidade na melhora dos alertas gerados pela aplicação.

Para a proposta de melhoria, foi necessária a atualização nas respostas providas pelos serviços das aplicações. A seguir, foram divididas as mudanças realizadas no *backend* e no *frontend*.

Backend:

- **Identificação de erros:** Identificar, de forma de clara, qual tipo de erro aconteceu.
- **Repassar o erro identificado:** Repassar o erro, identificado com mais detalhes.

Frontend:

- **Ícones de carregamento:** Apresenta ícones ao usuário, que indicam o sistema está processando.
- **Alertas em erros do sistema:** Apresenta uma descrição de erro ao usuário quando ocorrer um erro no sistema.

As melhorias de estado de sistema foram implementadas para os dois jogos, pois ambos apresentaram os mesmos problemas.

Para o jogo UnH, são implementados mais dois mecanismos além dos apresentados anteriormente, para que a dupla também implementasse mecanismos diferentes dos selecionados no jogo PDG. A seguir, estão apresentadas as propostas de melhorias para os mecanismos **entrada de texto estruturada** e **execução passo a passo** no jogo UnH.

6.2.4 Entrada de Texto Estruturada

Os problemas encontrados na Tabela 125 apresentam a necessidade de uma descrição informativa sobre o preenchimento do campo e suas limitações.

- **Informações sobre a entrada:** Apresentar de forma clara como preencher o campo e suas limitações.
- **Aviso:** Apresentar mudança visual e texto sobre inconformidade.

6.2.5 Execução Passo a Passo

Os problemas identificados no mecanismos Execução passo a passo, apresentados na Tabela 127 requerem informações de forma clara sobre como realizar o passo a passo de uma tarefa.

- **Informações passo a passo:** Apresentar de forma clara as informações de como realizar a tarefa.

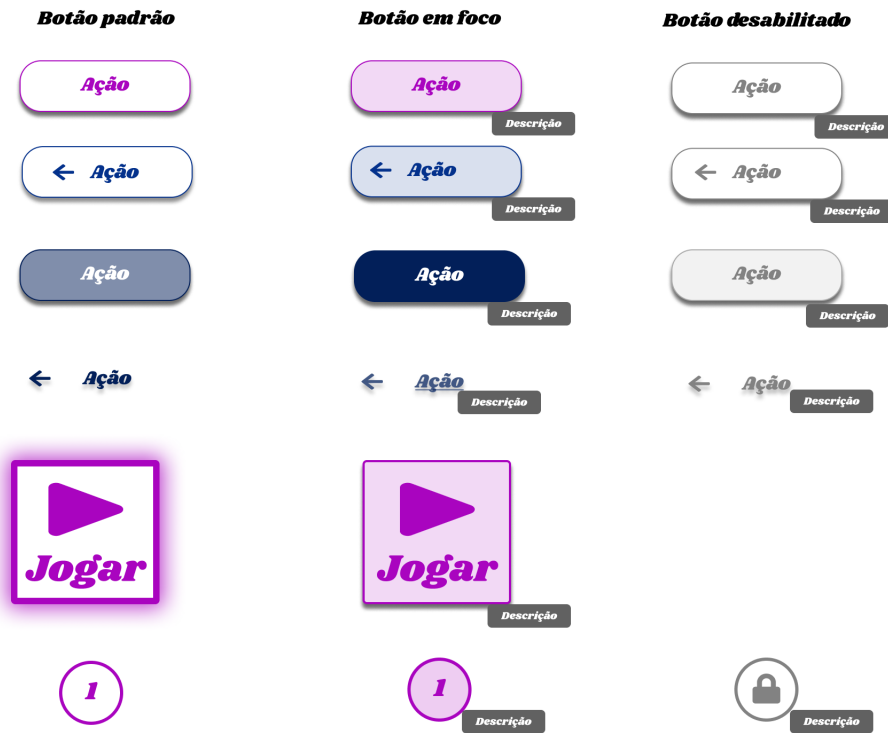
6.3 Modelagem

Nesta subseção serão apresentadas as modelagens propostas para as soluções descritas nas Subseções 6.2.1, 6.2.2 e 6.2.3.

Os protótipos gerados para as melhorias visam seguir ao máximo os padrões de *design* da interface, estabelecidos pelos desenvolvedores dos jogos. Para a construção dos mesmos, utilizou-se a ferramenta Figma.

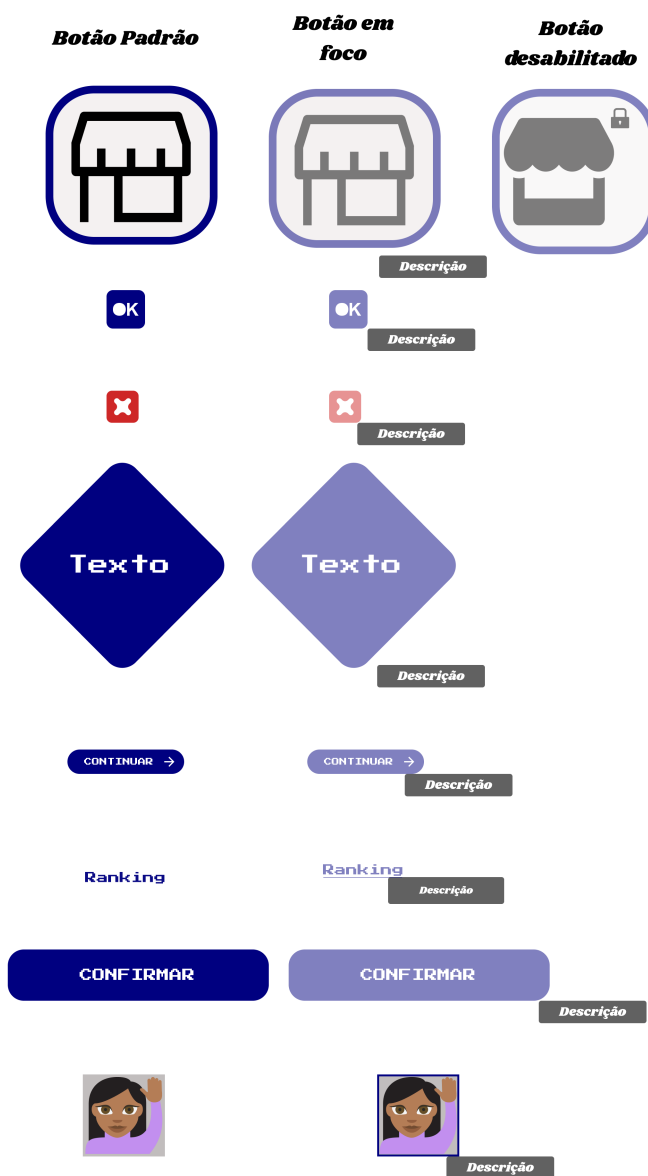
Para os botões dos jogos, foi criada uma padronização representada nas Figuras 8 e 9 através de um protótipo. Os botões contêm uma variação na opacidade e cor, também apresentam uma caixa de texto onde indica ao usuário a ação a ser gerada.

Figura 8 – Protótipo botões PDG



Fonte: Autores.

Figura 9 – Protótipo botões UnH



Fonte: Autores.

Os botões ao serem clicados podem apresentar um aviso, indicando que a ação do usuário foi aceita. A Figura 10 representa o alerta de solicitação aceita ao clicar em um botão.

Figura 10 – Protótipo alerta interação PDG



Fonte: Autores.

Figura 11 – Protótipo alerta interação UnH



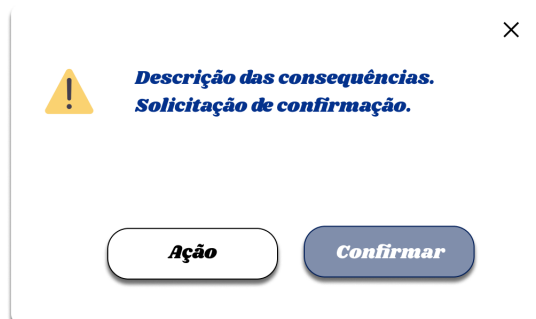
Fonte: Autores.

Os alertas deverão ser apresentados na parte superior direita da tela, onde descrevem uma informação que apresenta a confirmação do usuário ao realizar uma ação.

As propostas representadas nas Figuras 8 a 11 implementam as soluções listadas na Subseção 6.2.1, melhorando a interação. Deste modo, solucionando os problemas indicados nas Avaliações dos jogos (Capítulos 4 e 5).

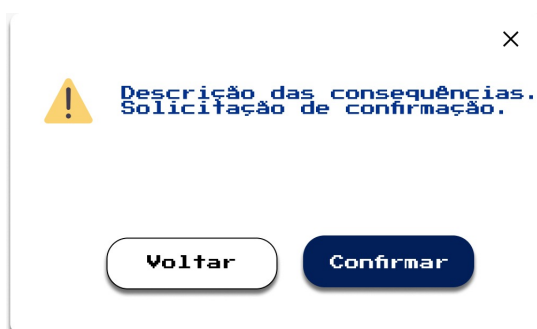
Para as correções dos problemas encontrados sobre o mecanismos de avisos, as Figuras 12 e 13 apresentam a solicitação de confirmação do usuário ao realizar a ação. A solicitação de confirmação contém um texto de descrição, onde pode ser indicado avisos sobre a tarefa.

Figura 12 – Protótipo alerta de confirmação PDG



Fonte: Autores.

Figura 13 – Protótipo alerta de confirmação UnH



Fonte: Autores.

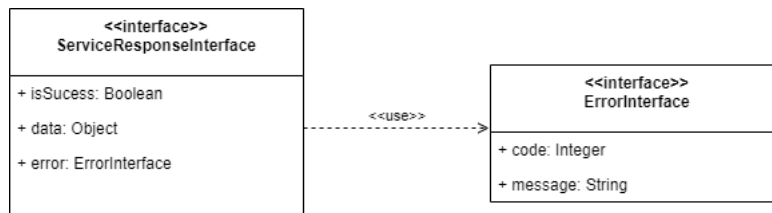
As **melhorias** propostas para a implementação do mecanismo de usabilidade **Estado do sistema** envolvem **atualizações** no código fonte dos **microserviços**, para que os avisos de falhas do sistema sejam apresentadas de maneira mais clara.

Os *backends* dos dois jogos não retornam de forma clara os erros do sistema, dificultando a apresentação de forma clara ao usuário os erros ocorridos no sistema. Os jogos possuem arquiteturas diferentes. Sendo assim, a padronização de retorno deve ser diferente entre os jogos.

Nos microserviços, do jogo PDG, (*Backend*) “persona-game-api” e “games-qa-api”, os *controllers* (controladores) recebem o retorno direto dos serviços, repassando os erros e dados de maneira arbitrária, dificultando o tratamento ao ocorrer erros. Como proposta de melhoria para este problema tem-se a padronização do retorno de todos os serviços, onde cada serviço deve retornar seguindo a estrutura estabelecida por uma interface.

A interface de retorno deve conter o tipo de estado, indicando o sucesso ou não da execução do serviço. Os erros retornados também seguiram uma estrutura padrão, contendo uma mensagem e um código para o erro. Com isso, é possível criar erros com mensagens personalizadas, específicas para cada serviço. A estrutura de retorno e erro são representadas no diagrama representado na Figura 14.

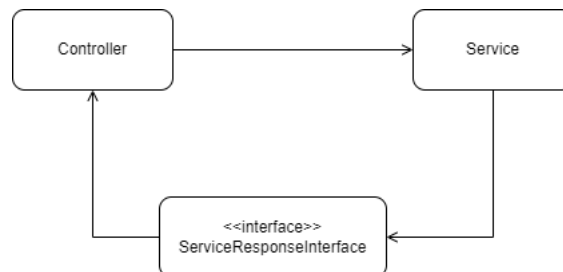
Figura 14 – Diagrama de classe resposta dos serviços



Fonte: Autores.

As *controllers* (controladores) receberão, por padrão dos serviços, objetos com a estrutura definida pela interface *ServiceResponseInterface*, representada na Figura 15.

Figura 15 – Diagrama padronização retorno serviço para *controller*



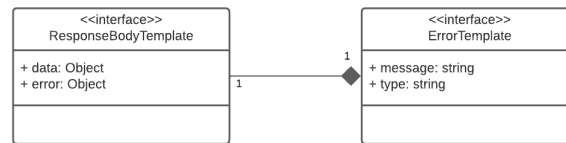
Fonte: Autores.

A estruturação realizada implementa o princípio de inversão de dependência do SOLID, e o padrão de projeto comportamental *template method* (Método padrão), onde é possível ter maior controle sobre o retorno dos serviços.

No jogo UnH a arquitetura é mais simplificada, onde no *controller* são realizadas as chamadas aos métodos do repositório que realiza a comunicação com banco de dados. Portanto, no controller é retornado o status e erro genérico provido pela biblioteca *typeorm*.

Para cada método presente na *controller*, o retorno do corpo da requisição é padronizado, contendo os dados da requisição ou erro com descrição. A Figura 16 apresenta a padronização proposta.

Figura 16 – Interface de padronização dos erros UnH



Fonte: Autores.

Com a implementação dessas melhorias nos microsserviços *Backend*, é possível retornar ao cliente (*Frontend*) as possíveis falhas do sistema de maneira clara e personalizada. No *Frontend*, é possível apresentar as falhas para o usuário através dos componentes modais, representadas nas Figuras 17 e 18.

Figura 17 – Protótipo alerta de falha no sistema PDG



Fonte: Autores.

Figura 18 – Protótipo alerta de falha no sistema UnH



Fonte: Autores.

Para representar que o sistema está carregando ou processando uma solicitação, foi prototipada uma sequência de ícones que representa o processamento do sistema, sendo o mesmo para ambos os jogos representados na Figura 19.

Figura 19 – Protótipo carregamento sistema.

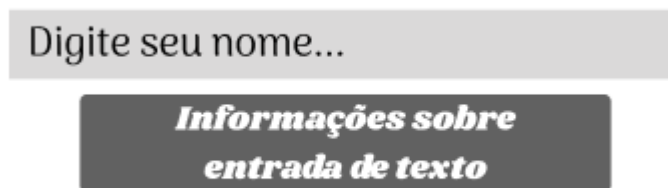


Fonte: Autores.

As propostas representadas nos diagramas (Figuras 14, 15 e 16) e as prototipações (Figuras 17 18 e 19) corrigem os problemas descritos para o **mecanismo estado do sistema**.

Para implementação de ajuda ao usuário em entradas de textos e informações sobre limitações do campo, foi prototipada uma descrição para o campo de entrada de textos, apresentado na Figura 20. Em caso de entrada inválida, também foi prototipado um alerta de erro, ilustrado na Figura 21, com uma descrição clara do erro do usuário.

Figura 20 – Protótipo descrição entrada de texto



Fonte: Autores.

Figura 21 – Protótipo alerta erro em campo de texto



Fonte: Autores.

Os problemas apontados na Seção 6.2.4 são corrigidos com as prototipações apresentadas nas Figuras 20 e 21. Desta forma, implementando as correções para o mecanismo entrada de texto estruturada.

6.4 Resumo do Capítulo

Neste capítulo, os mecanismos de usabilidade em que as melhorias foram propostas são selecionados. Além disso, melhorias foram apresentadas desde a modelagem à prototipação dos componentes, especificando onde as implementações levam à melhora no mecanismo em foco.

7 Implementação das Melhorias nos Jogos

Neste capítulo, os resultados das implementações das melhorias dos mecanismos de usabilidade selecionados foram apresentados. As implementações seguem o planejamento indicado no Capítulo 6.

As seções 7.1 à 7.5 deste capítulo apresentam as melhorias realizadas nos componentes dos dois jogos de acordo com o mecanismo em foco. Sendo assim, imagens com o estado anterior dos componentes sem melhorias e componentes atualizados são expostos (melhorados ou criados). Na seção 7.6 é abordada a condução da dupla durante o desenvolvimento, expondo a organização, dificuldades e desafios encontrados.

7.1 Interação

O mecanismo de usabilidade Interação, da família do *FeedBack*, tem como objetivo: “Informar o usuário do sistema que uma interação do usuário foi registrada“, como apresentado no Tabela 2.2. As melhorias realizadas trazem para o usuário mais informações sobre a interação realizada com os jogos.

As Figuras 22 e 24 apresentam os componentes atuais dos jogos PDG e UnH sem as melhorias. As melhorias na interação destes componentes estão apresentadas nas Figuras 23 e 27, de acordo com o planejado na Seção 6.3.

Figura 22 – PDG - Interação sem melhorias



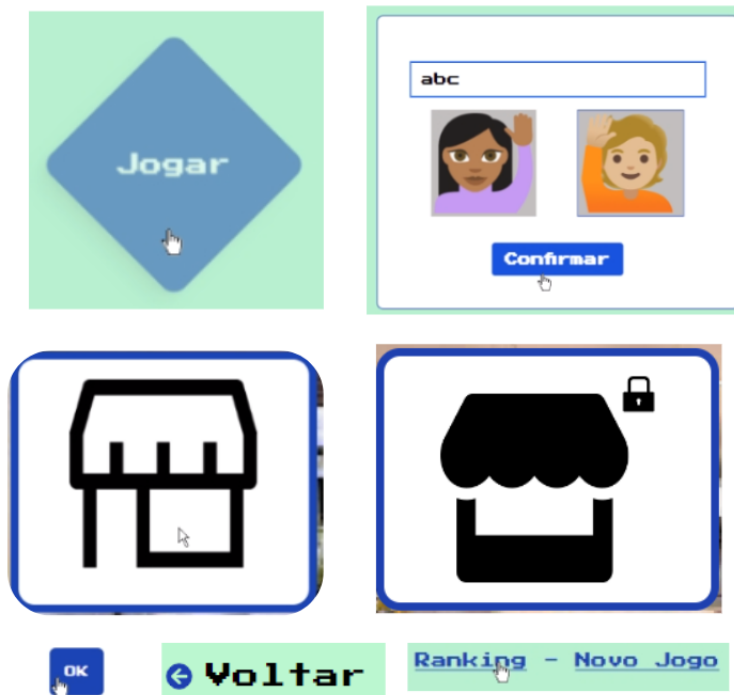
Fonte: Autores.

Figura 23 – PDG - Interação com melhorias



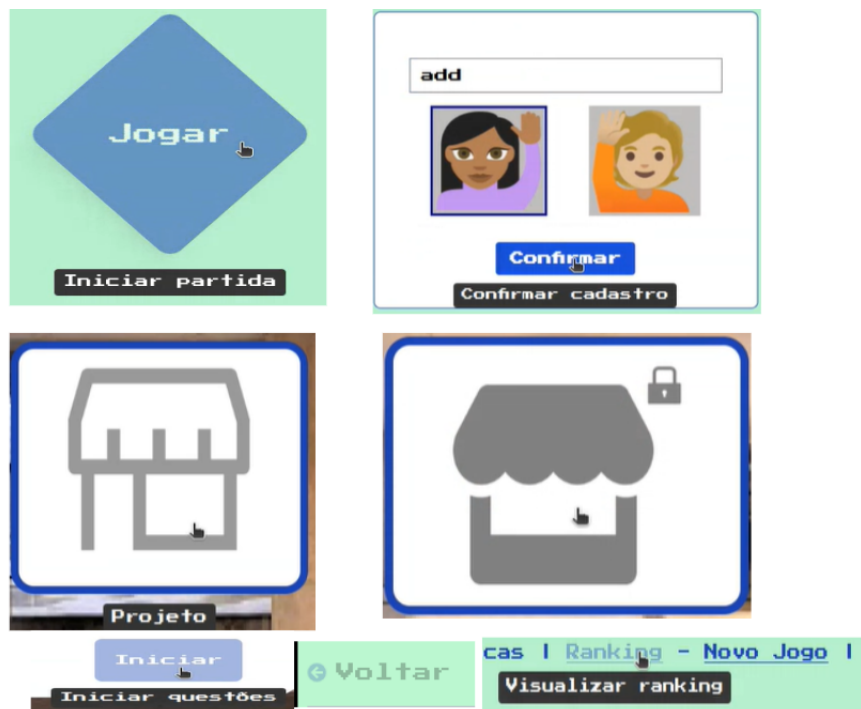
Fonte: Autores.

Figura 24 – UnH - Interação sem melhorias



Fonte: Autores.

Figura 25 – UnH - Interação com melhorias



Fonte: Autores.

As melhorias na interação ocorreram nos botões, *cards* (cartões) e *links*, implementando o que foi prototipado nas Figuras 8 e 9 da Seção 6.3.

Os componentes dos jogos, apresentados nas Figuras 23 e 27, sofrem alteração na opacidade quando estão em foco, textos informativos aos botões e mudança de coloração dos botões quando desabilitados. Portanto, dando mais destaque e informações aos componentes, ajudando o usuário de forma visual sobre como interagir.

7.2 Aviso

O Aviso é um dos mecanismos da família do *Feedback*, possuindo como objetivo: “Informar o usuário sobre ação com importantes consequências“, apresentado na Tabela 2.2. Portanto, as melhorias realizadas tiveram como foco a implementação de componentes de alertas, para indicar que resultado da ação pode acarretar perda de dados e/ou progresso.

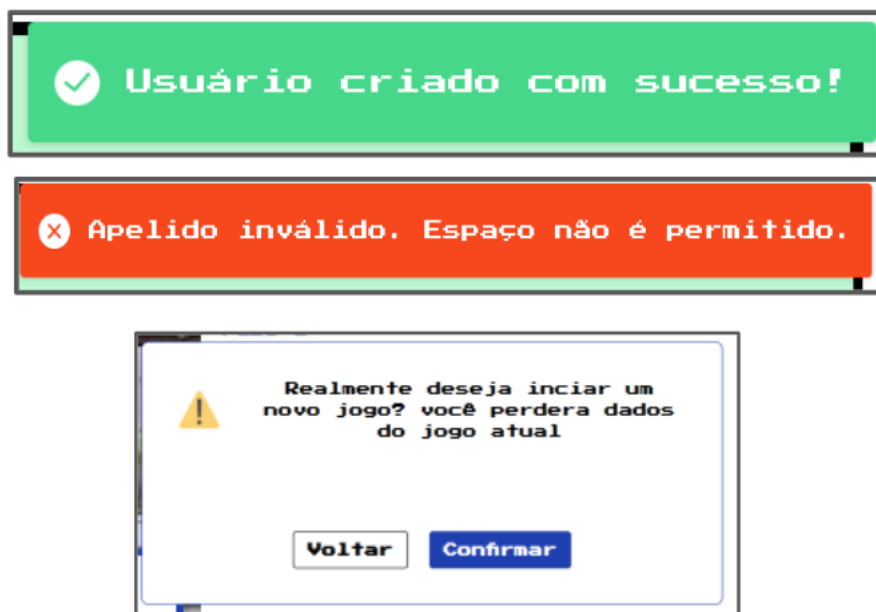
Os jogos PDG e UnH não possuíam componentes de avisos ao usuário, sendo necessária a implementação destes componentes. As Figuras 26 e 27 mostram os novos componentes implementados para o mecanismo aviso, como o planejado na Seção 6.3.

Figura 26 – PDG - Aviso com melhorias



Fonte: Autores.

Figura 27 – UnH - Aviso com melhorias



Fonte: Autores.

Para os dois jogos, foram adicionados os componentes de aviso, notificando o usuário sobre interação realizada e a coleta da confirmação da ação, casos onde pode haver perda de dados e/ou progresso. Os componentes prototipados estão apresentados nas Figuras 26 e 27 da Seção 6.3.

Foram adicionados alertas de interação e de confirmações das ações nos jogos, apresentados nas Figuras 26 e 27, avisando o usuário sobre suas interações e informações de consequências das ações.

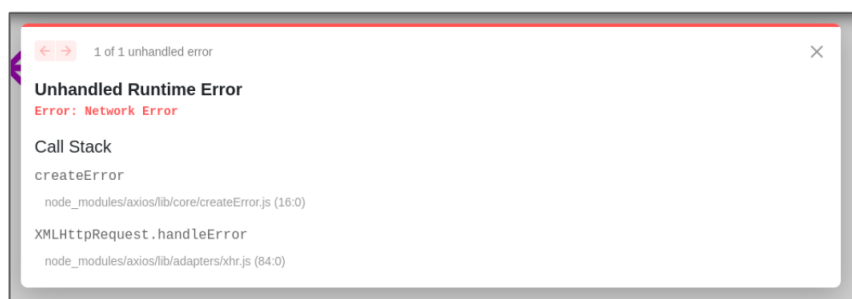
7.3 Estado do Sistema

O Estado do sistema da família do *Feedback* visa “Manter o usuário informado sobre o estado do sistema“, apresentado na Tabela 2.2. Sendo assim, as melhorias para a implementação deste mecanismo têm como foco apresentar ao usuário informações sobre como o sistema se comportou com a interação feita, como erros e exceções.

A apresentação do estado do sistema não estava de forma clara ou não estava sendo realizada nos jogos. Portanto, sendo necessária sua implementação. A Figura 28 apresenta o aviso sobre erro no sistema no PDG, não sendo claro ao usuário a informação. A Figura 29 apresenta a melhoria deste componente no PDG.

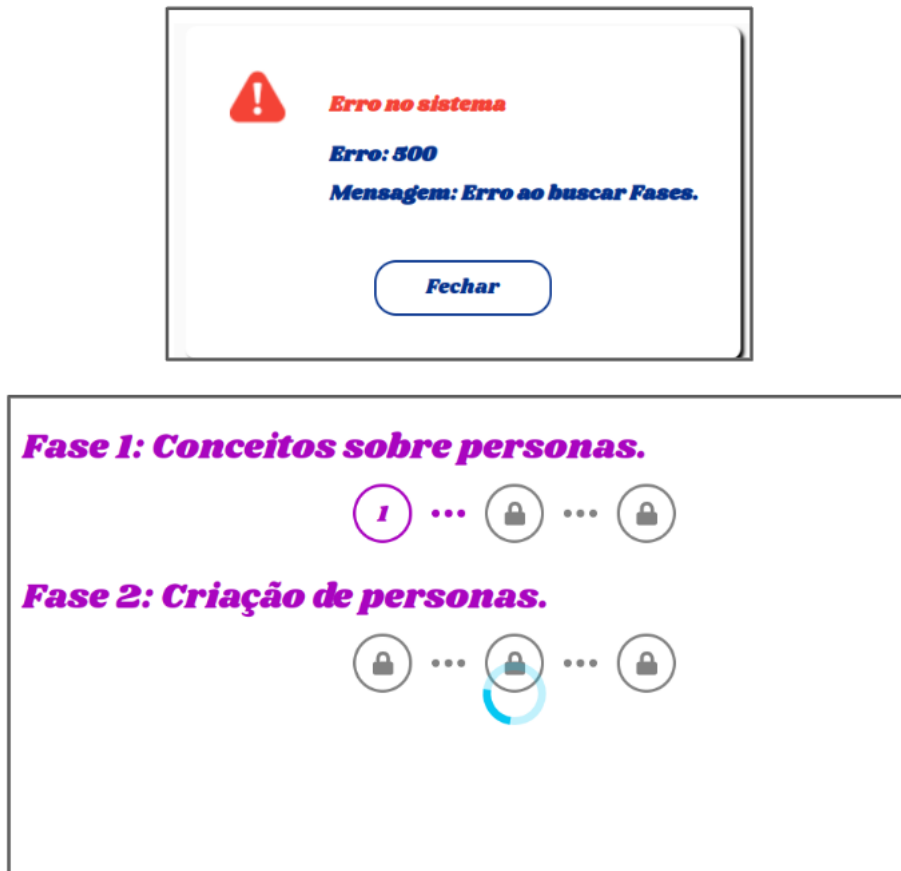
O jogo UnH não possuía componente de aviso sobre erros no sistema. Sendo assim, a Figura 30 apresenta a implementação deste componente.

Figura 28 – PDG - Estado do sistema antes da melhoria



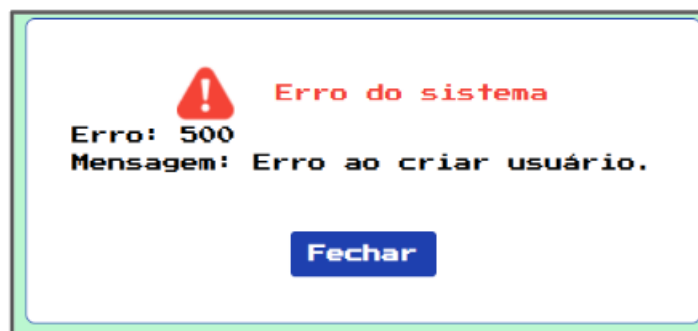
Fonte: Autores.

Figura 29 – PDG - Estado do sistema com melhoria



Fonte: Autores.

Figura 30 – UnH - Estado do sistema com melhoria



Fonte: Autores.

As Figuras 29 e 30 mostram as melhorias realizadas sobre os avisos do estado do sistema. As melhorias apresentam de forma mais clara sobre o erro ocorrido, trazendo o código do erro e a mensagem sobre o que ocorreu padronizado pelos serviços de *backend*. Além disso, o ícone de carregamento (*loading*) para indicar que o sistema está processando a informação também foi adicionado.

As prototipações destes componentes, que implementam o mecanismo de estado do sistema, podem ser encontradas nas Figuras 17, 18 e 19. No jogo UnH, não foi necessário a inclusão do ícone de carregamento, pois o jogo possui um tempo de carregamento breve. Desta forma, o ícone de carregamento não ficaria visível para o usuário.

7.4 Entrada de Texto Estruturada

O mecanismo Entrada de texto estruturada, da família Prevenção/correção de erros de entrada de usuário, tem como foco “Prevenção/correção de erros de entrada de usuário”, como apresentado na Tabela 2.2. O jogo UnH requer que o usuário adicione um apelido para cadastro. Portanto, a melhoria implementada visa dar mais informações ao usuário para o preenchimento correto deste campo.

O jogo UnH não apresentava as informações sobre limitações para preenchimento de campos de textos. A Figura 31 exibe os componentes que implementam o mecanismo Entrada de texto estruturada. No jogo PDG, não foi planejado a implementação deste mecanismo por conta de o jogo não possuir entradas de texto.

Figura 31 – UnH - Entrada de texto estruturada com melhoria



Fonte: Autores.

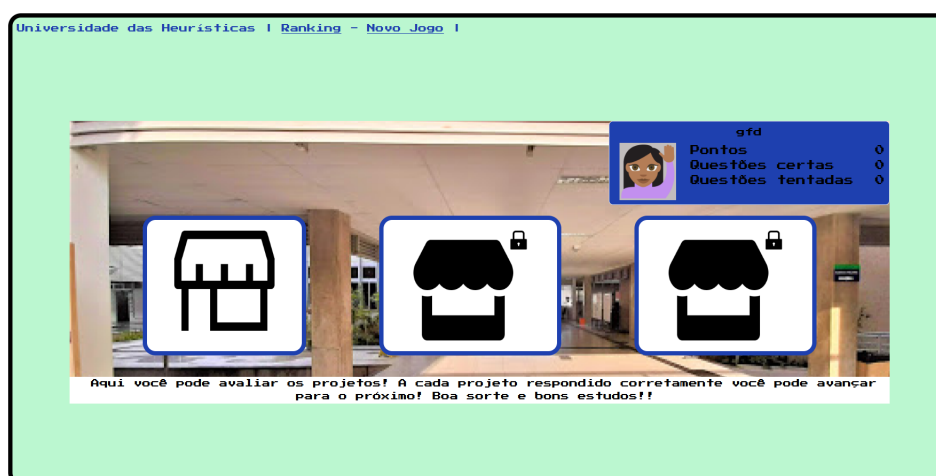
Como apresentado na Figura 31, foi adicionada informação sobre como preencher o campo de texto e mudanças visuais sobre inconformidades encontradas nos dados inseridos. Portanto, auxiliando o usuário na inserção de forma correta dos dados.

7.5 Execução Passo a Passo

O mecanismo Execução passo a passo, da família Assistente, tem como finalidade “Ajudar usuários a realizar tarefas que contém diferentes passos com entradas do usuário e corrigir tal entrada”, conforme descrito na Tabela 2.2. A atualização de textos informativos, complementado com as alterações realizadas para o mecanismo de interação, orientam o usuário para o cumprimento de sua tarefa.

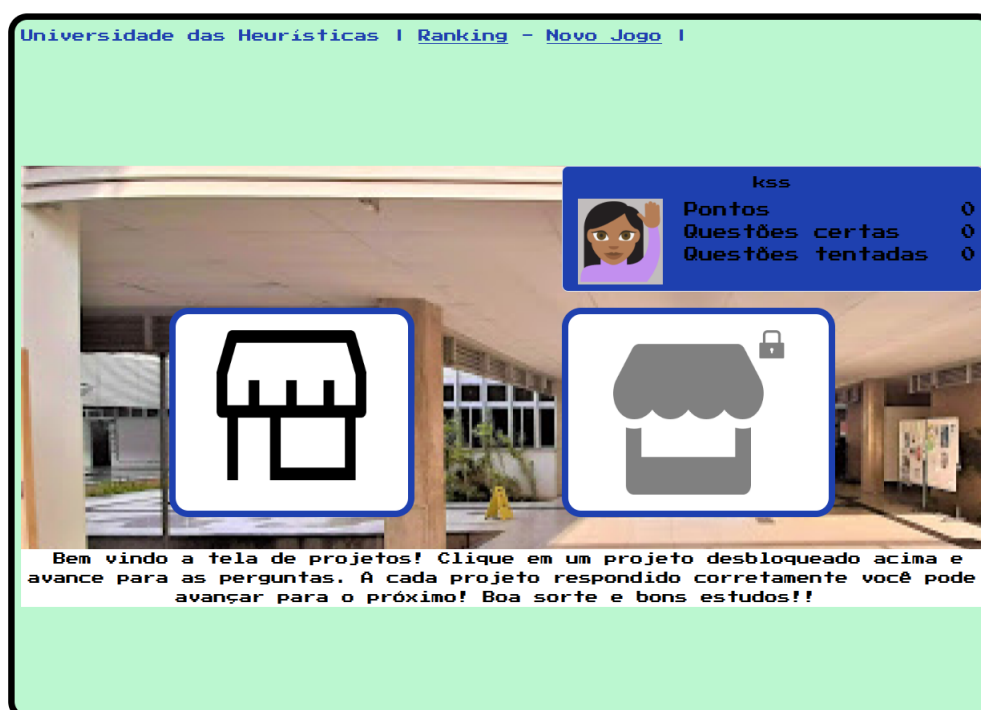
Informações sobre como realizar a tarefa em algumas páginas, como a de projetos, não estavam claras. Mudanças sutis nos textos, juntamente com a implementação dos outros mecanismos melhora a indicação sobre como a tarefa pode ser realizada. A figura 33 apresenta a melhoria para início das fases no jogo UnH.

Figura 32 – UnH - Execução passo a passo sem melhoria



Fonte: Autores.

Figura 33 – UnH - Execução passo a passo com melhoria



Fonte: Autores.

Na Figura 33, está apresentada a mudança no texto para que as informações da execução passo a passo fiquem mais claras.

7.6 Condução do Desenvolvimento

Nesta seção são descritos como foi realizada a organização da dupla e do código gerado, como se deu o desenvolvimento e dificuldades encontradas nas melhorias dos jogos.

Nos repositórios github, em que estão armazenados os códigos fontes dos jogos, foi criada a *branch* (ramo) “melhorias-mecanismos-usabilidade“, originada da *main* (principal). Com a *branch* de trabalho criada, a dupla iniciou as implementações das melhorias planejadas na Seção 6.3.

Semanalmente, os autores se reuniam, realizando um pareamento, para não somente desenvolver as melhorias, mas discutir sobre o desenvolvido. Desta forma, tentando seguir e/ou verificar a possibilidade de um replanejamento do que foi pensado.

As aplicações foram construídas com base na linguagem de programação JavaScript, sendo essa linguagem já conhecida e utilizada pelos autores. Portanto, pode-se entender mais rapidamente as regras utilizadas nas funcionalidades dos jogos.

A biblioteca React Js, utilizada para a construção das interfaces gráficas do PDG, já era conhecida pelos autores. Sendo assim, a dupla teve maior facilidade para a construção dos componentes e ações no jogo PDG. Entretanto, o jogo UnH utiliza o *framework* Vue.js, onde a dupla teve seu primeiro contato, levando a uma maior dificuldade de compreensão e implementação.

Além das dificuldades com o *framework* Vue.JS, as aplicações realizavam a utilização de eventos, o que dificultou a captura de algumas respostas provindas das APIs *backend*. Ao emitir os eventos, o bloco de instrução não aguarda o evento e os dados de retorno. Portanto, dificultou na coleta do que foi solicitado e seu tratamento.

Apesar das dificuldades encontradas em algumas tecnologias durante o desenvolvimento, as melhorias foram implementadas como planejadas. Desta forma, os mecanismos com problemas, selecionados, foram implementados nos jogos PDG e UnH.

7.7 Resumo do Capítulo

Neste capítulo, os resultados das melhorias foram apresentados, indicando a versão anterior e a versão aprimoradas dos componentes. Cada melhoria realizada nos jogos foi descrita, especificando em que ponto se diferencia da versão anterior. Além disso, informações sobre a organização e o desenvolvimento foram apresentadas.

8 Conclusão

Neste trabalho, melhorias foram propostas na usabilidade de jogos digitais, com foco no ensino e na aprendizagem. As melhorias na usabilidade têm como foco a implementação de mecanismos de usabilidade para jogos.

Para propor as melhorias na usabilidade, utilizou-se de conceitos fundamentais na área de IHC, na Seção 2.1. Estes conceitos são importantes para a implementação e a avaliação de sistemas computacionais interativos para uso humano, como dito por [Hewett et al. \(1992\)](#).

A usabilidade foi um conceito abordado na Seção de IHC (Seção 2.1), sendo este tema o foco principal do trabalho. Na Seção 2.2, sobre mecanismos de usabilidade, a usabilidade é expressada sendo decomposta em características. Cada uma das características da usabilidade possui um conjunto de subtipos, nos quais foram nomeadas por [Juristo, Moreno e Sanchez-Segura \(2007\)](#), como mecanismos de usabilidade. Desta forma, atingindo o objetivo **conhecer os conceitos sobre mecanismos de usabilidade (OE1)** (Subseção 1.2).

Aplicar melhorias na usabilidade dos jogos é um dos objetivos deste trabalho, implementando alguns dos mecanismos de usabilidade. Os jogos, apresentados nas Seções 2.3.1 e 2.3.2, são jogos educacionais que abordam temas da disciplina IHC.

Na Seção 3, a metodologia empregada no trabalho está detalhada, desde a classificação da pesquisa até como os dados devem ser coletados. Com isso, foi definido que os jogos devem ser avaliados utilizando métodos de inspeção (Avaliação Heurística), para identificar problemas na usabilidade em relação aos mecanismos de usabilidade.

Os jogos PDG e UnH foram avaliados por meio de inspeção seguindo a metodologia definida na Seção 3.3. Os resultados encontrados foram revisados. No PDG (Seção 4) foi possível realizar a revisão em grupo, trazendo a visão de uma dupla de especialista que trabalharam com o mesmo tema (Mecanismos de Usabilidade). Na avaliação do UnH (Seção 5), não foi possível realizar a revisão dos resultados em grupo. Portanto, manteve a revisão em dupla dos autores.

As revisões em dupla e em grupo tiveram um papel importante para a identificação de falhas e suas correções, melhorando a confiabilidade dos dados gerados. A cada revisão feita, calculou-se o grau de consenso dos resultados obtidos por meio do índice Kappa, buscando alcançar o índice de concordância mínimo indicado na metodologia.

Com a finalização da avaliação dos jogos, obteve-se a classificação (Seções 3.3 e 4) dos mecanismos de usabilidade, identificando qual o grau de implementação ou se

sua aplicação não é necessária. Além disso, um conjunto de problemas na usabilidade foram identificados, sendo esses problemas relacionados a cada mecanismo de usabilidade. Sendo assim, o objetivo de **Identificar necessidades de melhorias na usabilidade em jogos digitais, com foco nos mecanismos de usabilidade (OE02)** (Subseção 1.2) foi alcançado.

A classificação dos mecanismos indicou que vários dos mecanismos de usabilidade da família *feedback* não estavam sendo totalmente implementados nos jogos. Portanto, os mecanismos desta família foram selecionados para elaboração de propostas de melhorias. No UnH, outros dois mecanismos além da família do *feedback* foram selecionado para melhoria. Desta forma, atingido o objetivo específico de **selecionar o mecanismos de usabilidade que serão implementados (OE03)**, descrito na Subseção 1.2.

As melhorias corrigem as violações encontradas na aplicação, implementam nos jogos os mecanismos de usabilidade selecionados. No Capítulo 6, foram arquitetadas melhorias através de protótipos e diagramas. Portanto, foi possível atingir o objetivo específico de **propor e implementar melhorias de usabilidade aos jogos (OE04)**, Subseção 1.2.

O **objetivo geral** do trabalho foi alcançado (Subseção 1.2), pois as melhorias geradas aprimoram a usabilidade dos jogos, por meio da implementação de soluções referentes aos mecanismos de usabilidade. O objetivo geral foi alcançado totalmente com a implementação das soluções de usabilidade nos jogos.

A implementação de melhorias aos jogos PDG e UnH, com foco na usabilidade, proporcionou a melhora nas visões a respeito da construção de softwares com foco na interação com o usuário. Os mecanismos de usabilidade era um termo desconhecido pelos autores. Desta forma, o maior desafio foi compreender e aplicar os conceitos, avaliando e projetando melhorias para dois jogos.

Os conhecimentos adquiridos e aplicados, durante o trabalho e todo curso de Engenharia de Software, acrescenta de maneira positiva para as carreiras profissionais e acadêmicas, melhorando a qualidade de aplicações que serão projetadas e construídas.

8.1 Futuras Evoluções

Os jogos PDG e UnH tiveram melhorias implementadas, neste projeto, com base nos mecanismos de usabilidade. A partir deste trabalho é possível a realização de evoluções, nesta seção são apresentada as evoluções identificadas.

A realização de testes de usabilidade para validar a melhora na usabilidade com a implementação dos mecanismos. Sendo assim, demonstrando que com a implementação dos mecanismos de usabilidade há uma melhora na usabilidade, como apresentado por

[Juristo, Moreno e Sanchez-Segura \(2007\)](#).

Outro ponto para evolução seria a implementação dos outros mecanismos, não implementado ou implementados parcialmente, nos jogos. Desta forma, seria aplicada melhorias em outros mecanismos das famílias: cancelar/desfazer, Prevenção/correção de erros de entrada de usuário, Perfil do usuário e Ajuda.

O processo de avaliação utilizado pode ser estudado e aperfeiçoado para averiguar a possibilidade da utilização deste método como uma ferramenta de ensino.

Referências

- ABNT, A. B. D. N. T. *NBR ISO/IEC 9126-1*. Rio de Janeiro: ABNT, 2003. Normas ABNT de Engenharia de software – Qualidade de produto – Parte 1: Modelo de qualidade. Citado 3 vezes nas páginas 25, 33 e 34.
- BARBOSA; JUNQUEIRA, S. D. *Interação humano-computador*. Rio de Janeiro: Elsevier, 2010. Citado 5 vezes nas páginas 25, 33, 46, 47 e 56.
- BETHKE, E. *Game development and production*. [S.l.], 2003. Citado na página 38.
- CAPILLA, R. et al. *Usability implications in software architecture: The case study of a mobile app*. [S.l.], 2020. Citado na página 40.
- CORAM, T.; LEE, L. *Experiences: A Pattern Language for User Interface Design*. [S.l.], 1996. [Http://www.maplefish.com/todd/papers/experiences/Experiences.html](http://www.maplefish.com/todd/papers/experiences/Experiences.html). Citado na página 106.
- FALKEMBACH, G. A. M. *O lúdico e os jogos educacionais*. [S.l.], 2006. Citado 2 vezes nas páginas 25 e 37.
- FILHO, E. J. M. A.; FILHO, M. C. F. *Planejamento da Pesquisa Científica, 2ª edição*. [S.l.], 2015. Grupo GEN, 2015. 9788522495351. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788522495351/>. Acesso em: 19 fev. 2022. Citado na página 43.
- GEE, J. P. *Bons videogames e boa aprendizagem*. [S.l.], 2009. 167-178 p. Disponível em: <https://periodicos.ufsc.br/index.php/perspectiva/article/view/2175-795X.2009v27n1p167/14515>. Citado na página 25.
- GIL, A. *Como elaborar projetos e pesquisa*. Paulo: Atlas, 1995. 3a ed. Citado na página 44.
- HEWETT et al. *ACM SIGCHI Curricula for Human-Computer Interaction*. [S.l.], 1992. ACM SIGCHI Report, ACM, NY. Disponível em: <http://old.sigchi.org/cd>. Citado 2 vezes nas páginas 33 e 95.
- HIX, D.; HARTSON, H. *Developing User Interfaces: Ensuring Usability Through Product and Process*. New York, NY, 1993. John Wiley & Sons. Citado na página 33.
- JUNIOR, R. P. *PersonaDesignGame (PDG): Um Jogo Educacional sobre Personas*. [S.l.], 2021. Citado 4 vezes nas páginas 39, 45, 143 e 144.
- JURISTO, N.; MORENO, A.; SANCHEZ-SEGURA, M.-I. *Usability Elicitation Patterns (USEPs)*. [S.l.], 2006. [Http://www.grise.upm.es/sites/extras/2/](http://www.grise.upm.es/sites/extras/2/). Citado 8 vezes nas páginas 34, 47, 105, 106, 107, 108, 109 e 110.
- JURISTO, N.; MORENO, A. M.; SANCHEZ-SEGURA, M.-I. *Guidelines for Eliciting Usability Functionalities*. [S.l.], 2007. IEEE T. Software Eng 33, 11: 744-758. Citado 7 vezes nas páginas 25, 34, 35, 36, 37, 95 e 97.

- LANDIS, J.; KOCH, G. *The measurement of observer agreement for categorical data*. [S.l.], 1977. 159-74 p. Citado 3 vezes nas páginas 48, 49 e 143.
- LUNA, E. R. et al. *Incorporating usability requirements in a test/model-driven web engineering approach*. [S.l.], 2010. Journal of Web Engineering v. 9, n. 2, pp. 132- 156. Citado na página 40.
- MORAN, T. *The Command Language Grammars: a representation for the user interface of interactive computer systems*. New York, NY, 1981. International Journal of Man-Machine Studies 15. Citado na página 33.
- NIELSEN, J. *Finding usability problems through heuristic evaluation*. [S.l.], 1992. Proceedings of ACM CHI'92. Citado na página 47.
- NIELSEN, J. *Usability Engineering*. New York, 1993. NY: Academic Press. Citado na página 25.
- NORMAN, D.A. *Psychology of Everyday Things*. [S.l.], 1988. Basic Books. Citado na página 33.
- OLIVEIRA, C.; NAVES, L. *Universidade das Heurísticas: Um jogo para ensino-aprendizagem das heurísticas de Nielsen*. [S.l.], 2021. Citado 5 vezes nas páginas 37, 38, 45, 144 e 145.
- PMI, P. M. I. *A Guide to the Project Management Body of Knowledge – PMBOK®*. [S.l.], 2017. Sexta edição. | Newtown Square, PA: Project Management Institute, 2017. | Série: Guia PMBOK | Inclui referências. Citado na página 27.
- PRATES, R.; BARBOSA, S.; SOUZA, C. de. *A case study for evaluating interface design through communicability*. [S.l.], 2000. Proceedings of the ACM International Conference on Designing Interactive Systems, DIS 2000, pp. 308–317. Citado na página 34.
- PRENSKY, M. *Aprendizagem baseada em jogos digitais*. [S.l.], 2021. Citado na página 37.
- RODRIGUEZ, F. D.; ACUÑA, S. T.; JURISTO, N. *Design and programming patterns for implementing usability functionalities in web applications*. [S.l.], 2015. J. Syst. Software 104, 1: 107-124. Citado na página 40.
- SENA, S. de et al. *Aprendizagem baseada em jogos digitais: a contribuição dos jogos epistêmicos na geração de novos conhecimentos*. [S.l.], 2016. Citado na página 37.
- SOUZA, C. de; BARBOSA, S. D. J. *A case study for evaluating interface design through communicability*. [S.l.], 2005. F. Paterno & V. Wulf (eds.), End User Development. Springer. Citado na página 34.
- SOUZA, C. de; LEITÃO, C. *A case study for evaluating interface design through communicability*. [S.l.], 2009. In: J.M. Carroll (ed.) Synthesis Lectures on Human-Centered Informatics. Princeton, NJ: Morgan & Claypool Publishers. Citado na página 34.
- TIBOCHA, F. D. R. *Obtención y Uso de Patrones para la Implementación de Funcionalidades de Usabilidad en Aplicaciones Web*. [S.l.], 2014. Tese de Doutorado. ETSI Informatica. Citado na página 46.

TIDWELL, J. *Designing Interfaces. Patterns for Effective Interaction Design*. [S.l.], 2005. O'Reilly. Citado na página 105.

USABILITY Pattern Collection. [S.l.], 2007.
[Http://www.cmis.brighton.ac.uk/research/patterns/home.html](http://www.cmis.brighton.ac.uk/research/patterns/home.html). Citado na página 106.

WELIE, M. van. *The Amsterdam Collection of Patterns in User Interface Design*. [S.l.], 2007. [Http://www.welie.com](http://www.welie.com). Citado na página 106.

Apêndices

APÊNDICE A – Mecanismos de Usabilidade-Tabelas de Apoio

Tabela 23 – *Feedback*: Estado do sistema

Identificação
Nome: Estado do sistema
Família: <i>Feedback</i>
Contexto
<p>Quando alguma alteração no estado do sistema ocorre, o usuário deve ser notificado (TIDWELL, 2005), especialmente quando a mudança de estado afeta as informações de estado que provavelmente mudarão ao longo do tempo (por exemplo, barra de estado em aplicativos de janelas, trem, ônibus ou horário de companhias aéreas, displays de videocassete...)</p> <p>Quando alterações importantes para o usuário ocorrem ou</p> <p>Quando ocorrem falhas importantes para o usuário:</p> <ul style="list-style-type: none"> - Durante a execução da tarefa - Porque não há recursos suficientes do sistema <p>Porque os recursos externos não estão funcionando corretamente.</p>
Questão(ões)
<ol style="list-style-type: none"> 1. Os estado do sistema é informado ao usuário? 2. Em caso de falha do sistema, o usuário é notificado? 3. O sistema de software fornece <i>feedback</i> sobre os recursos em caso de falhas?

Fonte: Adaptado de Juristo, Moreno e Sanchez-Segura (2006)

Tabela 24 – *Feedback*: Interação

Identificação
Nome: Interação
Família: <i>Feedback</i>
Contexto
Quando o usuário realiza um evento de interação, como o mouse clique, movimento do mouse, movimento de seta, prensa de teclado, etc. o sistema deve informar ao usuário que a interação foi aceita (CORAM; LEE, 1996)
Questão(ões)
1. Ao realizar eventos com o mouse e/ou teclado o sistema informa ao usuário sobre a interação?

Fonte: Adaptado de Juristo, Moreno e Sanchez-Segura (2006)

Tabela 25 – *Feedback*: *Feedback* para ação longa

Identificação
Nome: <i>Feedback</i> para ação longa
Família: <i>Feedback</i>
Contexto
Quando um processo demorado interrompe a interface do usuário por dois segundos ou mais.
Questão(ões)
1. Em tarefas com tempo de espera maior que 2 segundos o usuário é notificado sobre a ação longa? 2. As informações sobre as ações longas são atualizadas a cada 2 segundos?

Fonte: Adaptado de Juristo, Moreno e Sanchez-Segura (2006)

Tabela 26 – *Feedback*: Aviso

Identificação
Nome: Aviso
Família: <i>Feedback</i>
Contexto
Quando uma ação que tem sérias consequências foi exigida pelo usuário (USABILITY..., 2007; WELIE, 2007).
Questão(ões)
1. O usuário é informado sobre as consequências da execução da tarefa? 2. O usuário é avisado de formas diferente dependendo do grau de consequências da ação?

Fonte: Adaptado de Juristo, Moreno e Sanchez-Segura (2006)

Tabela 27 – Desfazer/Cancelar: Desfazer global

Identificação
Nome: Desfazer global
Família: Desfazer/Cancelar
Contexto
Ao construir um sistema altamente interativo com funcionalidades múltiplas e complexas
Questão(ões)
1. Em tarefas de múltiplas etapas é possível desfazer/reverter as etapas? 2. As etapas em tarefas são gravadas possibilitando ser desfeitas?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 28 – Desfazer/Cancelar: Desfazer específico do objeto

Identificação
Nome: Desfazer específico do objeto
Família: Desfazer/Cancelar
Contexto
Ao construir um sistema altamente interativo com funcionalidades múltiplas e complexas em objetos específicos do sistema
Questão(ões)
1. Em em um objeto específico em uma tarefa é possível desfazer/reverter?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 29 – Desfazer/Cancelar: Abortar Operação

Identificação
Nome: Abortar Operação
Família: Desfazer/Cancelar
Contexto
Quando o usuário precisa sair de um aplicativo ou um comando rapidamente.
Questão(ões)
1. A aplicação possibilita ao usuário a saída de forma clara e imediata? 2. A tarefa possui mecanismos de cancelamento para aborta-las? 3. Ações que levam mais de 10 segundo podem ser abortadas?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 30 – Desfazer/Cancelar: Voltar

Identificação
Nome: Voltar
Família: Desfazer/Cancelar
Contexto
Quando há aplicativos interativos com várias etapas.
Questão(ões)
1. A aplicação possibilita ao usuário a opção de voltar a estados anteriores nas etapas? 2. A aplicação possibilita ao usuário a opção de voltar a um estado de segurança?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 31 – Prevenção/correção de erros de entrada do usuário: Entrada de Texto Estruturado

Identificação
Nome: Entrada de Texto Estruturado
Família: Prevenção/correção de erros de entrada do usuário
Contexto
Quando o sistema só pode aceitar entradas do usuário em um formato muito específico
Questão(ões)
1. O sistema de software fornece ao usuário informações sobre como os dados devem ser inseridos?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 32 – Assistente: Execução Passo a Passo

Identificação
Nome: Execução Passo a Passo
Família: Assistente
Contexto
Quando um usuário não especialista precisa executar uma tarefa complexa pouco frequente, composta por várias sub-tarefas, onde as decisões precisam ser tomadas em cada sub-tarefa.
Questão(ões)
1. Ao realizar tarefas com varias etapas, informações são fornecidas sobre cada etapa?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 33 – Perfil de Usuário: Preferências

Identificação
Nome: Preferências
Família: Perfil de Usuário
Contexto
Quais informações precisam ser obtidas e especificadas para fornecer aos usuários o mecanismo de “preferências”.
Questão(ões)
<ol style="list-style-type: none"> 1. O usuário consegue configurar preferências particulares a aplicação(fontes, cores, idioma, etc.)? 2. A aplicação provê um conjunto de preferências de acordo com perfis de usuários?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 34 – Perfil de Usuário: Espaço de Objetos Pessoais

Identificação
Nome: Espaço de Objetos Pessoais
Família: Perfil de Usuário
Contexto
Quais informações precisam ser obtidas e especificadas para fornecer aos usuários o mecanismo de “espaço de objeto pessoal”.
Questão(ões)
<ol style="list-style-type: none"> 1. O usuário tem a possibilidade de organizar seu próprio layout? 2. O sistema permite que isso seja feito para cada usuário?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 35 – Perfil de Usuário: Favoritos

Identificação
Nome: Favoritos
Família: Perfil de Usuário
Contexto
Em um sistema de software navegável, quando o sistema é possivelmente grande e complexo e permite que o usuário se mova livremente por ele de maneiras não suportadas diretamente pela estrutura do artefato
Questão(ões)
<ol style="list-style-type: none"> 1. O usuário consegue gravar os lugares(telas) que mais visitou ou que possui maior preferência?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 36 – Ajuda: Ajuda Multinível

Identificação
Nome: Ajuda Multinível
Família: Ajuda
Contexto
Quando o aplicativo a ser desenvolvido é complexo e alguns usuários provavelmente precisarão de um sistema de ajuda completo, mas a maioria dos usuários não terá tempo para usá-lo; então, os desenvolvedores querem dar suporte a usuários impacientes e/ou ocasionais.
Questão(ões)
1. O sistema fornece ajuda ao usuário em tarefas? 2. Ao realizar a tarefa fornece dicas de como realiza-lá?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

Tabela 37 – Agregação de Comandos

Identificação
Nome: Agregação de Comandos
Família: Agregação de Comandos
Contexto
Quando os usuários precisam repetir longas sequências de ações e quando as possíveis ações a serem realizadas com o artefato podem ser expressas por meio de comandos, que podem ser compostos de partes menores, em uma sintaxe semelhante a uma linguagem com regras precisas e apreensíveis, e os usuários são dispostos e capazes de aprender essa sintaxe.
Questão(ões)
1. Em ações complexas exigem comandos agregados? 2. O usuário consegue nomear um macro?

Fonte: Adaptado de [Juristo, Moreno e Sanchez-Segura \(2006\)](#)

B Avaliação PDG

B.1 Revisão em dupla

Tabela 38 – Inspeção - Estado do sistema

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O estado do sistema é informado ao usuário?	Parcialmente implementado	Parcialmente implementado	Parcialmente implementado
Em caso de falha do sistema, o usuário é notificado?	Parcialmente implementado	Parcialmente implementado	Parcialmente implementado
O sistema de software fornece <i>feedback</i> sobre os recursos em caso de falhas?	Não se Aplica	Não se Aplica	Não se Aplica

Fonte: Autores.

Tabela 39 – Violação - Estado do sistema

Caso(s) de uso	UC01, UC02 e UC03		
Local	Em todo o sistema		
Problema(s)	Ao carregar as questões de uma fase, o estado do sistema é informado através de um ícone de maneira discreta. Ao ocorrer uma falha do sistema a mensagem não é clara para o usuário, erro genérico do <i>framework</i>		
Classificação	Parcialmente implementado	Figura(s)	Figura 34

Fonte: Autores.

Tabela 40 – Inspeção - Interação

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
Ao realizar eventos com o mouse e/ou teclado, o sistema informa ao usuário sobre a interação?	Parcialmente implementado	Parcialmente implementado	Implementado

Fonte: Autores.

Tabela 41 – Violação - Interação

Caso(s) de uso	UC01 e UC02		
Local	Em todo o sistema(Botões)		
Problema(s)	O sistema fornece uma ação visual em alguns eventos com o <i>mouse</i> , porém não contém um <i>feedback</i> explicativo da ação, como mudança visual no componente ou texto explicativo.		
Classificação	Parcialmente implementado	Figura(s)	Figura 35

Fonte: Autores.

Tabela 42 – Inspeção - *Feedback* para ações longas

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
Em tarefas com tempo de espera maior que 2 segundos o usuário é notificado sobre a ação longa?	Não se aplica	Não se aplica	Não se aplica
As informações sobre as ações longas são atualizadas a cada 2 segundos?	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Para o mecanismo de usabilidade, “***Feedback para ações longas***” não se aplica, pois no sistema não possui ações com tempo de processamento maior que 2 segundos. Não possuindo uma tabela de violação.

Tabela 43 – Inspeção - Aviso

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O usuário é informado sobre as consequências da execução da tarefa?	Não Implementado	Não se aplica	Não se aplica
O usuário é avisado de formas diferentes dependendo do grau de consequências da ação?	Não Implementado	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 44 – Violação - Aviso

Caso(s) de uso	UC01		
Local	Em todo o sistema(Botões)		
Problema(s)	Não é informado ao usuário a consequência das ações ao realizar uma tarefa. Não possui diferentes níveis de aviso como ao sair de uma fase ou voltar para a questão anterior		
Classificação	Não Implementado	Figura(s)	Figura 36

Fonte: Autores.

Nos casos de uso, UC02 e UC03 não se aplica, pois não a tarefas que geram consequências ao usuário.

Tabela 45 – Inspeção - Desfazer global

Questão(ões)	Casos de uso		
	UC1	UC2	UC3
Em tarefas de múltiplas etapas é possível desfazer/reverter as etapas?	Não se aplica	Não se aplica	Não se aplica
As etapas em tarefas são gravadas possibilitando ser desfeitas?	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

O Mecanismo de usabilidade “**desfazer global**” não se aplica, pois o sistema é bem simples não possuindo tarefas complexas, onde uma etapa deve ser desfeita/revertida. Não possui tabela de violação.

Tabela 46 – Inspeção - Desfazer específico do objeto.

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
Em em um objeto específico em uma tarefa é possível desfazer/reverter?	Não implementado	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 47 – Violação - Desfazer específico do objeto

Caso(s) de uso	UC01		
Local	Tela de questões		
Problema(s)	Nas questões é possível voltar para questão anterior, porem não não é possível refazê-la		
Classificação	Não Implementado	Figura(s)	

Fonte: Autores.

Nos casos de uso UC02 e UC03, não possui tarefa que pode ser revertida ou cancelada.

Tabela 48 – Inspeção - Abortar Operação

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
A aplicação possibilita ao usuário a saída de forma clara e imediata?	Implementado	Implementado	Implementado
A tarefa possui mecanismos de cancelamento e confirmação da ação (abortar)?	Não Implementado	Não se aplica	Não se aplica
Ações que levam mais de 10 segundo podem ser abortadas?	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 49 – Violação - Abortar Operação

Caso(s) de uso	UC01		
Local	Em todo o sistema (Botões cancelar e/ou sair)		
Problema(s)	Não possui um diálogo de confirmação do aborto da operação.		
Classificação	Parcialmente implementado	Figura(s)	Figura 36

Fonte: Autores.

Tabela 50 – Inspeção - Voltar

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
A aplicação possibilita ao usuário a opção de voltar a estados anteriores nas etapas?	Parcialmente implementado	Implementado	Implementado
A aplicação possibilita ao usuário a opção de voltar a um estado de segurança?	Implementado	Implementado	Implementado

Fonte: Autores.

Tabela 51 – Violação - Voltar

Caso(s) de uso	UC01		
Local	Em todo o sistema		
Problema(s)	Ao voltar para a questão anterior o estado da questão não é mantido, sendo necessário refazê-la.		
Classificação	Parcialmente implementado	Figura(s)	

Fonte: Autores.

Tabela 52 – Inspeção - Preferências

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O usuário consegue configurar preferências particulares na aplicação (fontes, cores, idioma, etc.)?	Não Implementado	Não Implementado	Não Implementado
A aplicação provê um conjunto de preferências de acordo com perfis de usuários?	Não Implementado	Não Implementado	Não Implementado

Fonte: Autores.

Tabela 53 – Violação - Preferências

Caso(s) de uso	UC01, UC02 e UC03		
Local	Em todo o sistema		
Problema(s)	A aplicação não realiza a coleta de dados do usuário, o usuário utilizado é gerado automaticamente pela plataforma, não possibilitando determinar e gravar a preferência do mesmo.		
Classificação	Não implementado	Figura(s)	

Fonte: Autores.

Tabela 54 – Inspeção - Espaço de Objetos Pessoais

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O usuário tem a possibilidade de organizar seu próprio <i>layout</i> ?	Não Implementado	Não Implementado	Não Implementado
O sistema permite que isso seja feito para cada usuário?	Não Implementado	Não Implementado	Não Implementado

Fonte: Autores.

Tabela 55 – Violação - Espaço de Objetos Pessoais

Caso(s) de uso	UC01, UC02 e UC03		
Local	Em todo o sistema		
Problema(s)	A aplicação não realiza a coleta de dados do usuário. O usuário utilizado é gerado automaticamente pela plataforma, não possibilitando determinar e gravar a objetos pessoais do mesmo.		
Classificação	Não Implementado	Figura(s)	

Fonte: Autores.

Tabela 56 – Inspeção - Favoritos

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O usuário consegue gravar os lugares (telas) que mais visitou ou que possui maior preferência?	Não Implementado	Não Implementado	Não Implementado

Fonte: Autores.

Tabela 57 – Violação - Favoritos

Caso(s) de uso	UC01, UC02 e UC03		
Local	Em todo o sistema		
Problema(s)	A aplicação não realiza a coleta de dados do usuário. O usuário utilizado é gerado automaticamente pela plataforma, não possibilitando determinar e gravar os favoritos do mesmo.		
Classificação	Não implementado	Figura(s)	

Fonte: Autores.

Tabela 58 – Inspeção - Ajuda multinível

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O sistema fornece ajuda ao usuário em tarefas?	Não Implementado	Não se aplica	Não se aplica
Ao realizar a tarefa fornece dicas de como realizá-la?	Não Implementado	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 59 – Violação - Ajuda multinível

Caso(s) de uso	UC01		
Local	Tela de fases e questões		
Problema(s)	O sistema não oferece ajuda a diferentes tipos de usuário, como dicas para execução da tarefa.		
Classificação	Não Implementado	Figura(s)	

Fonte: Autores.

O mecanismo Ajuda multinível não se aplica aos casos de uso UC02 e UC03, pois são tarefas simples que não necessitam de ajuda.

Tabela 60 – Inspeção - Agregação de Comandos

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
Em ações complexas existem comandos agregados?	Não se aplica	Não se aplica	Não se aplica
O usuário consegue nomear um macro?	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

A aplicação PersonaDesignGame não possui uma longa sequência de ações em que o usuário precisa repeti-las. Deste modo, não se aplica a avaliação deste mecanismos.

Tabela 61 – Inspeção - Execução Passo a Passo

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
Ao realizar tarefas com varias etapas, informações são fornecidas sobre cada etapa?	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

A aplicação não possui ações complexas, onde o passo a passo não possui diferentes entradas de usuários, que necessitam prover assistência ao usuário. Deste modo, não se aplica a avaliação deste mecanismo.

Tabela 62 – Inspeção - Entrada de texto estruturado

Questão(ões)	Casos de uso		
	UC01	UC02	UC03
O sistema de software fornece ao usuário informações sobre como os dados devem ser inseridos?	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

A aplicação não possui entradas de usuário para prevenção de erros como entradas de textos, ou ações complexas com mouse. Deste modo, não se aplica a avaliação deste mecanismo.

C Avaliação UnH

C.1 Avaliação individual: João

Nesta subseção, está apresentada a avaliação individual do membro João Pedro.

Tabela 63 – Inspeção - Estado do sistema

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O estado do sistema é informado ao usuário?	Não implementado	Não implementado	Não implementado	Não implementado
Em caso de falha do sistema, o usuário é notificado?	Não implementado	Não implementado	Não implementado	Não implementado
O sistema de software fornece <i>feedback</i> sobre os recursos em caso de falhas?	Não implementado	Não implementado	Não implementado	Não implementado

Fonte: João Pedro.

Tabela 64 – Violação - Estado do sistema

Caso(s) de uso	UC01, UC02, UC03 e UC04		
Local	Em todo o sistema		
Problema(s)	Não possui nenhuma informação sobre o status do sistema.		
Classificação	Não implementado	Figura(s)	Figura 34

Fonte: João Pedro.

Tabela 65 – Inspeção - Interação

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Ao realizar eventos com o mouse e/ou teclado, o sistema informa ao usuário sobre a interação?	Parcialmente implementado	Parcialmente implementado	Parcialmente implementado	Parcialmente implementado

Fonte: João Pedro.

Tabela 66 – Violação - Interação

Caso(s) de uso	UC01, UC02, UC03 e UC04		
Local	Em todo o sistema		
Problema(s)	O sistema fornece uma ação visual em alguns eventos com o <i>mouse</i> , porém não contém um <i>feedback</i> explicativo da ação, como mudança visual no componente ou texto explicativo.		
Classificação	Parcialmente implementado	Figura(s)	

Fonte: João Pedro.

Tabela 67 – Inspeção - *Feedback* para ações longas

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em tarefas com tempo de espera maior que 2 segundos o usuário é notificado sobre a ação longa?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
As informações sobre as ações longas são atualizadas a cada 2 segundos?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

Para o mecanismo de usabilidade, “Feedback para ações longas” não se aplica, pois no sistema não possui ações com tempo de processamento maior que 2 segundos. Não possuindo uma tabela de violação.

Tabela 68 – Inspeção - Aviso

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário é informado sobre as consequências da execução da tarefa?	Não Implementado	Não Implementado	Não se aplica	Não se aplica
O usuário é avisado de formas diferentes dependendo do grau de consequências da ação?	Não Implementado	Não Implementado	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 69 – Violação - Aviso

Caso(s) de uso	UC01 e UC02		
Local	Em todo o sistema		
Problema(s)	O usuário não é informado sobre as consequências geradas pela ação		
Classificação	Não Implementado	Figura(s)	

Fonte: João Pedro.

Não se aplica para os casos de uso UC03 e UC04, pois não geram consequências ao usuário.

Tabela 70 – Inspeção - Desfazer global

Questão(ões)	Casos de uso			
	UC1	UC2	UC3	UC4
Em tarefas de múltiplas etapas é possível desfazer/reverter as etapas?	Não Implementado	Não Implementado	Não se aplica	Não se aplica
As etapas em tarefas são gravadas possibilitando ser desfeitas?	Não Implementado	Não Implementado	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 71 – Violação - Desfazer global

Caso(s) de uso	UC01 e UC02		
Local	Em todo o sistema		
Problema(s)	Não é possível desfazer nenhuma ação no sistema		
Classificação	Não Implementado	Figura(s)	

Fonte: João Pedro.

Não se aplica para os casos de uso UC03 e UC04, pois não possuem ações que precisam ser desfeitas.

Tabela 72 – Violação - Desfazer específico do objeto

Caso(s) de uso	UC01		
Local	Em todo o sistema		
Problema(s)	Não é possível desfazer nenhuma ação no sistema		
Classificação	Não Implementado	Figura(s)	

Fonte: João Pedro.

Não se aplica para os casos de uso UC03 e UC04, pois não possuem ações que precisam ser desfeitas.

Tabela 73 – Inspeção - Desfazer específico do objeto.

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em em um objeto específico em uma tarefa é possível desfazer/reverter?	Não Implementado	Não Implementado	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 74 – Violação - Desfazer específico do objeto

Caso(s) de uso	UC01 e UC02		
Local	Em todo o sistema		
Problema(s)	Não é possível desfazer nenhuma ações específicas no sistema		
Classificação	Não Implementado	Figura(s)	

Fonte: João Pedro.

Não se aplica para os casos de uso UC03 e UC04, pois não possuem ações específicas que precisam ser desfeitas.

Tabela 75 – Inspeção - Abortar Operação

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
A aplicação possibilita ao usuário a saída de forma clara e imediata?	Não Implementado	Implementado	Não se aplica	Não se aplica
A tarefa possui mecanismos de cancelamento e confirmação da ação (abortar)?	Não implementado	implementado	Não se aplica	Não se aplica
Ações que levam mais de 10 segundo podem ser abortadas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 76 – Violação - Abortar Operação

Caso(s) de uso	UC01		
Local	Em todo o sistema		
Problema(s)	Não é possível abordar a operação de criar usuário.		
Classificação	Parcialmente implementado	Figura(s)	

Fonte: João Pedro.

Não se aplica para os casos de uso UC03 e UC04, pois não possuem ações que precisam ser desfeitas.

Tabela 77 – Inspeção - Voltar

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
A aplicação possibilita ao usuário a opção de voltar a estados anteriores nas etapas?	Não implementado	Não implementado	Implementado	Implementado
A aplicação possibilita ao usuário a opção de voltar a um estado de segurança?	Não implementado	Não implementado	Implementado	Implementado

Fonte: João Pedro.

Tabela 78 – Violação - Voltar

Caso(s) de uso	UC01 e UC02		
Local	Botões		
Problema(s)	Não é possível voltar no cadastro de usuário. É possível voltar nas respostas de questões, porém a jogada será perdida.		
Classificação	Parcialmente implementado	Figura(s)	

Fonte: João Pedro.

Tabela 79 – Inspeção - Prevenção/correção de erros de entrada de usuário

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O sistema de software fornece ao usuário informações sobre como os dados devem ser inseridos?	Não implementado	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 80 – Violação - Prevenção/correção de erros de entrada de usuário

Caso(s) de uso	UC01		
Local	Tela cadastro de usuário		
Problema(s)	Não possui Prevenção/correção de erro para a entrada no nome do usuário.		
Classificação	Parcialmente implementado	Figura(s)	

Fonte: João Pedro.

Tabela 81 – Inspeção - Passo a Passo

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Ao realizar tarefas com varias etapas, informações são fornecidas sobre cada etapa?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

A aplicação não possui ações complexas onde o passo a passo não possui diferentes entradas de usuários, que necessitam prover assistência ao usuário. Deste modo não se aplica a avaliação deste mecanismos

Tabela 82 – Inspeção - Preferências

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário consegue configurar preferências particulares na aplicação (fontes, cores, idioma, etc.)?	Não implementado	Não implementado	Não implementado	Não implementado
A aplicação provê um conjunto de preferências de acordo com perfis de usuários?	Não implementado	Não implementado	Não implementado	Não implementado

Fonte: João Pedro.

Tabela 83 – Violação - Preferências

Caso(s) de uso	UC01, UC02, UC03 E UC04		
Local	Todo o sistema		
Problema(s)	Não possui nenhum mecanismo que possibilite adicionar às preferências do usuário.		
Classificação	Não implementado	Figura(s)	

Fonte: João Pedro.

Tabela 84 – Inspeção - Espaço de Objetos Pessoais

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário tem a possibilidade de organizar seu próprio <i>layout</i> ?	Não implementado	Não implementado	Não implementado	Não implementado
O sistema permite que isso seja feito para cada usuário?	Não implementado	Não implementado	Não implementado	Não implementado

Fonte: João Pedro.

Tabela 85 – Violação - Espaço de Objetos Pessoais

Caso(s) de uso	UC01, UC02, UC03 E UC04		
Local	Todo o sistema		
Problema(s)	Não possui nenhum mecanismo que possibilite que o usuário organize o próprio layout de forma individual.		
Classificação	Não implementado	Figura(s)	

Fonte: João Pedro.

Tabela 86 – Inspeção - Favoritos

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário consegue gravar os lugares (telas) que mais visitou ou que possui maior preferência?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

Por se tratar de uma aplicação simples que não possui uma grande navegação no sistema o mecanismo de Favoritos não se aplica.

Tabela 87 – Inspeção - Ajuda multinível

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O sistema fornece ajuda ao usuário em tarefas?	Não implementado	Não implementado	Não se aplica	Não se aplica
Ao realizar a tarefa fornece dicas de como realizá-la?	Não implementado	Não implementado	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 88 – Violação - Ajuda multinível

Caso(s) de uso	UC01 e UC02		
Local	Tela cadastro de usuário e tela de questões		
Problema(s)	Informações de como realizar a tarefa está incompleta.		
Classificação	Não implementado	Figura(s)	

Fonte: João Pedro.

Tabela 89 – Inspeção - Agregação de Comandos

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em ações complexas existem comandos agregados?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
O usuário consegue nomear um macro?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

A aplicação não possui uma longa sequência de ações em que o usuário precisa repeti-las. Deste modo não se aplica a avaliação deste mecanismos.

C.2 Avaliação individual: Moacir

Nesta subseção, está apresentada a avaliação individual do membro Moacir Junior.

A Tabelas 90 à 111 apresentam os resultados da avaliação.

Tabela 90 – Inspeção - Estado do sistema

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O estado do sistema é informado ao usuário?	Não implementado	Não implementado	Não implementado	Não implementado
Em caso de falha do sistema, o usuário é notificado?	Não implementado	Não implementado	Não implementado	Não implementado
O sistema de software fornece <i>feedback</i> sobre os recursos em caso de falhas?	Não implementado	Não implementado	Não implementado	Não implementado

Fonte: Moacir Junior.

Tabela 91 – Violação - Estado do sistema

Caso(s) de uso	UC01, UC02, UC03 e UC04		
Local	Em todo o sistema		
Problema(s)	Ao realizar uma interação não é informado que a informação está sendo processada ou a apresentação de uma falha.		
Classificação	Não implementado	Figura(s)	Figura

Fonte: Moacir Junior.

Tabela 93 – Violação - Interação

Caso(s) de uso	UC02		
Local	Botões		
Problema(s)	Os cartões com as fases não apresentam nenhuma mudanças quando estão em foco.		
Classificação	Parcialmente implementado	Figura(s)	Figura 37

Fonte: Moacir Junior.

Tabela 92 – Inspeção - Interação

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Ao realizar eventos com o mouse e/ou teclado, o sistema informa ao usuário sobre a interação?	Parcialmente implemen- tado	Parcialmente implemen- tado	Parcialmente implemen- tado	Implementado

Fonte: Moacir Junior.

Tabela 94 – Inspeção - *Feedback* para ações longas

Questão(ões)	Casos de uso							
	UC01		UC02		UC03		UC04	
Em tarefas com tempo de espera maior que 2 segundos o usuário é notificado sobre a ação longa?	Não aplica	se	Não aplica	se	Não aplica	se	Não aplica	se
As informações sobre as ações longas são atualizadas a cada 2 segundos?	Não aplica	se	Não aplica	se	Não aplica	se	Não aplica	se

Fonte: Moacir Junior.

Não se aplica pois não há ações no jogo que ultrapassem 2 segundos para serem processadas.

Tabela 95 – Inspeção - Aviso

Questão(ões)	Casos de uso							
	UC01		UC02		UC03		UC04	
O usuário é informado sobre as consequências da execução da tarefa?	Não aplica	se	Não imple- mentado		Não aplica	se	Não aplica	se
O usuário é avisado de formas diferentes dependendo do grau de consequências da ação?	Não aplica	se	Não imple- mentado		Não aplica	se	Não aplica	se

Fonte: Moacir Junior.

Tabela 96 – Violação - Aviso

Caso(s) de uso	UC01		
Local	Em todo o sistema(Botões)		
Problema(s)	Não avisa que ao iniciar uma fase os dados serão salvos em seu usuário e que demandará muito tempo. Ao clicar em ver ranking ou novo jogo não é informado que os dados serão perdidos.		
Classificação	Não Implementado	Figura(s)	

Fonte: Moacir Junior.

Tabela 97 – Inspeção - Desfazer global

Questão(ões)	Casos de uso			
	UC1	UC2	UC3	UC4
Em tarefas de múltiplas etapas é possível desfazer/reverter as etapas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
As etapas em tarefas são gravadas possibilitando ser desfeitas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

O jogo não possui tarefas de múltiplas etapas que precisam ser desfeitas.

Tabela 98 – Inspeção - Desfazer específico do objeto.

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC05
Em em um objeto específico em uma tarefa é possível desfazer/reverter?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

O jogo não possui tarefas onde é possível desfazer ou reverter.

Tabela 99 – Inspeção - Abortar Operação

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
A aplicação possibilita ao usuário a saída de forma clara e imediata?	Não se aplica	Implementado	Não se aplica	Não se aplica
A tarefa possui mecanismos de cancelamento e confirmação da ação (abortar)?	Não implementado	Não implementado	Não se aplica	Não se aplica
Ações que levam mais de 10 segundo podem ser abortadas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

Tabela 100 – Violação - Abortar Operação

Caso(s) de uso	UC01 e UC02		
Local	Tela cadastro e Tela de questão		
Problema(s)	Não é dada a possibilidade ao usuário abordar/cancelar a operação.		
Classificação	Não implementado	Figura(s)	Figura 37

Fonte: Moacir Junior.

Tabela 101 – Inspeção - Voltar

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
A aplicação possibilita ao usuário a opção de voltar a estados anteriores nas etapas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
A aplicação possibilita ao usuário a opção de voltar a um estado de segurança?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Não se aplica pois não possui uma atividade de múltiplas etapas onde a etapa anterior precise de correções.

Tabela 102 – Inspeção - Prevenção/correção de erros de entrada de usuário

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O sistema de software fornece ao usuário informações sobre como os dados devem ser inseridos?	Parcialmente implementado	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

Tabela 103 – Violação - Prevenção/correção de erros de entrada de usuário

Caso(s) de uso	UC01		
Local	Tela cadastro		
Problema(s)	O número de cadastro é limitado, mas a descrição não informa. Também permite caracteres como espaços.		
Classificação	Parcialmente implementado	Figura(s)	

Fonte: Moacir Junior.

Tabela 104 – Inspeção - Passo a Passo

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Ao realizar tarefas com varias etapas, informações são fornecidas sobre cada etapa?	Não se aplica	Parcialmente implementado	Não se aplica	Não se aplica

Fonte: Moacir Junior.

Tabela 105 – Violação - Passo a Passo

Caso(s) de uso	UC02		
Local	Tela de projetos		
Problema(s)	Informações sobre como realizar a etapa não é indicado.		
Classificação	Parcialmente implementado	Figura(s)	Figura 37

Fonte: Moacir Junior.

Tabela 106 – Inspeção - Preferências

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário consegue configurar preferências particulares na aplicação (fontes, cores, idioma, etc.)?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
A aplicação provê um conjunto de preferências de acordo com perfis de usuários?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

A aplicação não coleta informações suficientes do usuário para este mecanismo. Não possui um perfil reutilizável.

Tabela 107 – Inspeção - Espaço de Objetos Pessoais

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário tem a possibilidade de organizar seu próprio <i>layout</i> ?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
O sistema permite que isso seja feito para cada usuário?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Não é algo idealizado para o jogo com simples funcionalidades. Os usuários possuem poucos dados não sendo reutilizável.

Tabela 108 – Inspeção - Favoritos

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário consegue gravar os lugares (telas) que mais visitou ou que possui maior preferência?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

Não é algo idealizado para o jogo com simples funcionalidades.

Tabela 109 – Inspeção - Ajuda multinível

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O sistema fornece ajuda ao usuário em tarefas?	Parcialmente implementado	Parcialmente implementado	Não se aplica	Não se aplica
Ao realizar a tarefa fornece dicas de como realizá-la?	Parcialmente implementado	Parcialmente implementado	Não se aplica	Não se aplica

Fonte: Moacir Junior.

Tabela 110 – Violação - Ajuda multinível

Caso(s) de uso	UC01 e UC02		
Local	Tela cadastro e projetos		
Problema(s)	Informações de como realizar a tarefa está incompleta.		
Classificação	Parcialmente implementado	Figura(s)	Figura 37 e 39

Fonte: Moacir Junior.

Tabela 111 – Inspeção - Agregação de Comandos

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em ações complexas existem comandos agregados?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
O usuário consegue nomear um macro?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Moacir Junior.

Jogo simples não possui sequência de ações longas onde seja necessária a agregação de comandos. Portanto, não se aplica.

C.3 Revisão em dupla

Nesta subseção, está apresentada a revisão em dupla do jogo UnH.

Tabela 112 – Inspeção - Estado do sistema

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O estado do sistema é informado ao usuário?	Não implementado	Não implementado	Não implementado	Não se aplica
Em caso de falha do sistema, o usuário é notificado?	Não implementado	Não implementado	Não implementado	Não implementado
O sistema de software fornece <i>feedback</i> sobre os recursos em caso de falhas?	Não implementado	Não implementado	Não implementado	Não implementado

Fonte: Autores.

Tabela 113 – Violação - Estado do sistema

Caso(s) de uso	UC01, UC02, UC03 e UC04		
Local	Em todo o sistema		
Problema(s)	Ao realizar uma interação não é informado que a informação está sendo processada ou a apresentação de uma falha.		
Classificação	Não implementado	Figura(s)	Figura 34

Fonte: Autores.

Tabela 114 – Inspeção - Interação

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Ao realizar eventos com o mouse e/ou teclado, o sistema informa ao usuário sobre a interação?	Não implementado	Não implementado	Não implementado	Não implementado

Fonte: Autores.

Tabela 115 – Violação - Interação

Caso(s) de uso	UC01, UC02, UC03 e UC04		
Local	Botões, entrada de texto e cartões		
Problema(s)	Alguns botões mudam de opacidade ao passar o mouse, porém não possuem confirmação de interação aceita. Alguns botões e cartões quando está em foco não apresentam mudanças visuais		
Classificação	Parcialmente implementado	Figura(s)	Figura 37

Fonte: Autores.

Tabela 116 – Inspeção - *Feedback* para ações longas

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em tarefas com tempo de espera maior que 2 segundos o usuário é notificado sobre a ação longa?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
As informações sobre as ações longas são atualizadas a cada 2 segundos?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Para o mecanismo de usabilidade, “*Feedback* para ações longas” não se aplica, pois no sistema não possui ações com tempo de processamento maior que 2 segundos.

Tabela 117 – Inspeção - Aviso

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário é informado sobre as consequências da execução da tarefa?	Não Implementado	Não Implementado	Não se aplica	Não se aplica
O usuário é avisado de formas diferentes dependendo do grau de consequências da ação?	Não Implementado	Não Implementado	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 118 – Violação - Aviso

Caso(s) de uso	UC01 e UC02		
Local	Em todo o sistema		
Problema(s)	Em ações onde os dados seriam gravados ou alterados e em tarefas que demandam muito tempo ao usuário, não informam as consequências das ações. O usuário não é informado sobre o grau de consequência das ações		
Classificação	Não Implementado	Figura(s)	

Fonte: Autores.

Tabela 119 – Inspeção - Desfazer global

Questão(ões)	Casos de uso			
	UC1	UC2	UC3	UC4
Em tarefas de múltiplas etapas é possível desfazer/reverter as etapas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
As etapas em tarefas são gravadas possibilitando ser desfeitas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Para o mecanismo “Desfazer específico de objeto” não se aplica, pois jogo não possui tarefas de múltiplas etapas que precisam ser desfeitas. Deste modo não se aplica para este mecanismo.

Tabela 120 – Inspeção - Desfazer específico do objeto.

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em em um objeto específico em uma tarefa é possível desfazer/reverter?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Para o mecanismo “Desfazer específico de objeto” não se aplica, pois o jogo não possui uma tarefa a ser desfeita ou revertida. Por se tratar de um jogo simples em que a funcionalidade principal responder questões não tem o intuito de ser desfeita. Deste modo não se aplica para este mecanismo.

Tabela 121 – Inspeção - Abortar Operação

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
A aplicação possibilita ao usuário a saída de forma clara e imediata?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
A tarefa possui mecanismos de cancelamento e confirmação da ação (abortar)?	Não implementado	Parcialmente implementado	Não se aplica	Não se aplica
Ações que levam mais de 10 segundo podem ser abortadas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 122 – Violação - Abortar Operação

Caso(s) de uso	UC01 e UC02		
Local	Formulário de cadastro e tela de questões		
Problema(s)	Não é possível abordar a operação de criar usuário. Possui a possibilidade de cancelamento das questões, porém não a confirmação do cancelamento .		
Classificação	Parcialmente implementado	Figura(s)	Figuras 37 e 39

Fonte: Autores.

Tabela 123 – Inspeção - Voltar

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
A aplicação possibilita ao usuário a opção de voltar a estados anteriores nas etapas?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
A aplicação possibilita ao usuário a opção de voltar a um estado de segurança?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Para o mecanismo “Voltar” não se aplica, pois não possui uma atividade de múltiplas etapas onde a etapa anterior possui um estado de segurança. Deste modo não se aplica para este mecanismo.

Tabela 124 – Inspeção - Prevenção/correção de erros de entrada de usuário

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O sistema de software fornece ao usuário informações sobre como os dados devem ser inseridos?	Não implementado	Não se aplica	Não se aplica	Não se aplica

Fonte: João Pedro.

Tabela 125 – Violação - Prevenção/correção de erros de entrada de usuário

Caso(s) de uso	UC01		
Local	Entrada de texto		
Problema(s)	O número de cadastro é limitado, mas a descrição não informa. Também permite caracteres como espaços.		
Classificação	Não implementado	Figura(s)	Figura 39

Fonte: Autores.

Tabela 126 – Inspeção - Passo a Passo

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Ao realizar tarefas com varias etapas, informações são fornecidas sobre cada etapa?	Não se aplica	Parcialmente implementado	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 127 – Violação - Passo a Passo

Caso(s) de uso	UC02		
Local	Tela de projetos		
Problema(s)	Informações sobre como realizar a etapa(iniciar um projeto) não é indicada.		
Classificação	Parcialmente implementado	Figura(s)	Figura 37

Fonte: Autores.

Tabela 128 – Inspeção - Preferências

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário consegue configurar preferências particulares na aplicação (fontes, cores, idioma, etc.)?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
A aplicação provê um conjunto de preferências de acordo com perfis de usuários?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

A aplicação não coleta informações suficientes do usuário para este mecanismo. Não possui um perfil reutilizável.

Tabela 129 – Inspeção - Espaço de Objetos Pessoais

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário tem a possibilidade de organizar seu próprio <i>layout</i> ?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
O sistema permite que isso seja feito para cada usuário?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

A aplicação não coleta informações suficientes do usuário para este mecanismo. Não possui um perfil reutilizável. Não é algo idealizado para o jogo com simples funcionalidades. Os usuários possuem poucos dados, não sendo reutilizados.

Tabela 130 – Inspeção - Favoritos

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O usuário consegue gravar os lugares (telas) que mais visitou ou que possui maior preferência?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

A aplicação não coleta informações suficientes do usuário para este mecanismo. Não possui um perfil reutilizável. Não é algo idealizado para o jogo com simples funcionalidades. Portanto, não se aplica.

Tabela 131 – Inspeção - Ajuda multinível

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
O sistema fornece ajuda ao usuário em tarefas?	Parcialmente implementado	Parcialmente implementado	Não se aplica	Não se aplica
Ao realizar a tarefa fornece dicas de como realizá-la?	Não Implementado	Parcialmente implementado	Não se aplica	Não se aplica

Fonte: Autores.

Tabela 132 – Violação - Ajuda multinível

Caso(s) de uso	UC01 e UC02		
Local	Tela de cadastro e tela de projetos		
Problema(s)	As informações de como realizar as tarefas estão incompletas.		
Classificação	Parcialmente implementado	Figura(s)	Figura 37 e 39

Fonte: Autores.

Tabela 133 – Inspeção - Agregação de Comandos

Questão(ões)	Casos de uso			
	UC01	UC02	UC03	UC04
Em ações complexas existem comandos agregados?	Não se aplica	Não se aplica	Não se aplica	Não se aplica
O usuário consegue nomear um macro?	Não se aplica	Não se aplica	Não se aplica	Não se aplica

Fonte: Autores.

Por se tratar de um jogo simples de perguntas e respostas, não possui sequências de ações longas que seja possível a agregação de comandos. Desta forma não se aplica para este mecanismo.

Anexos

ANEXO A – Primeiro Anexo

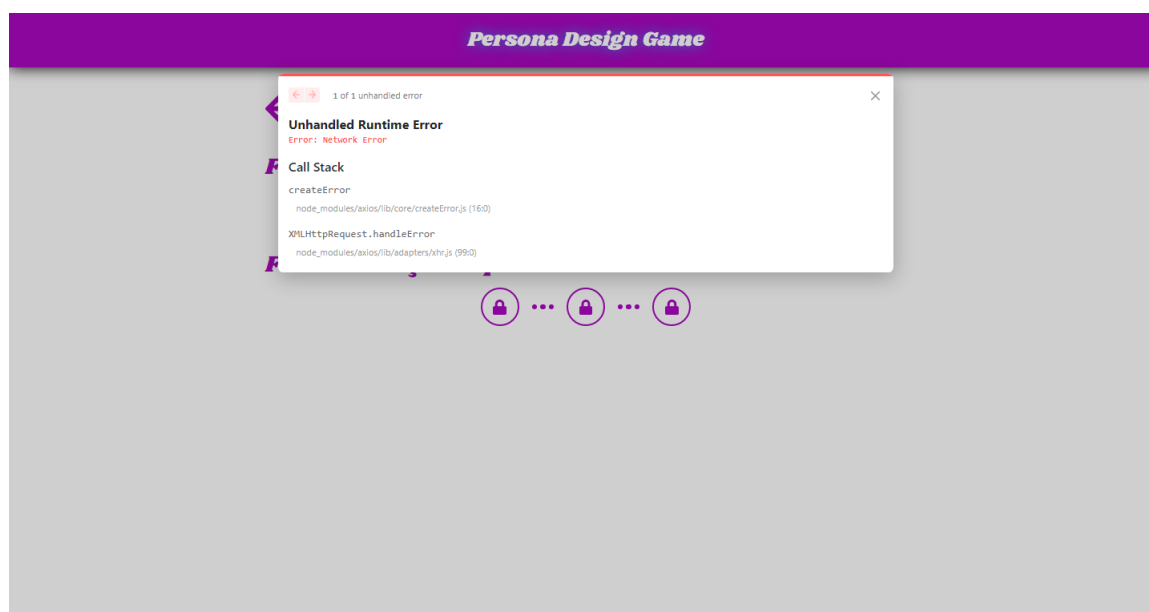
Este capítulo possui conteúdo de outros autores utilizados neste trabalho, apresentando tabelas e figuras de apoio.

Tabela 134 – Magnitude coeficiente Kappa

Coeficiente Kappa	Força da concordância
0	Pobre
0,01 - 0,2	Desprezível
0,21 - 0,4	Suave
0,41 - 0,6	Moderada
0,61 - 0,8	Substancial
0,81 - 1	Quase Perfeita

Fonte: Adaptado de [Landis e Koch \(1977\)](#)

Figura 34 – Falha no sistema.



Fonte: [Junior \(2021\)](#).

Figura 35 – Ação com o mouse.



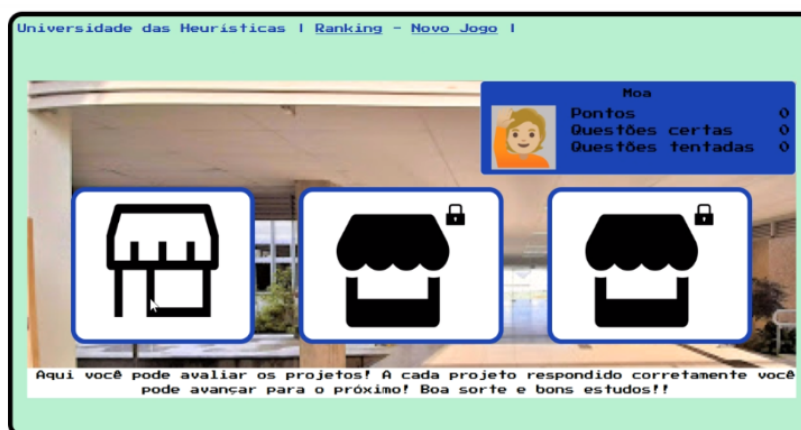
Fonte: Junior (2021).

Figura 36 – Falta de descrição da consequência da ação.



Fonte: Junior (2021).

Figura 37 – Tela projetos UnH.



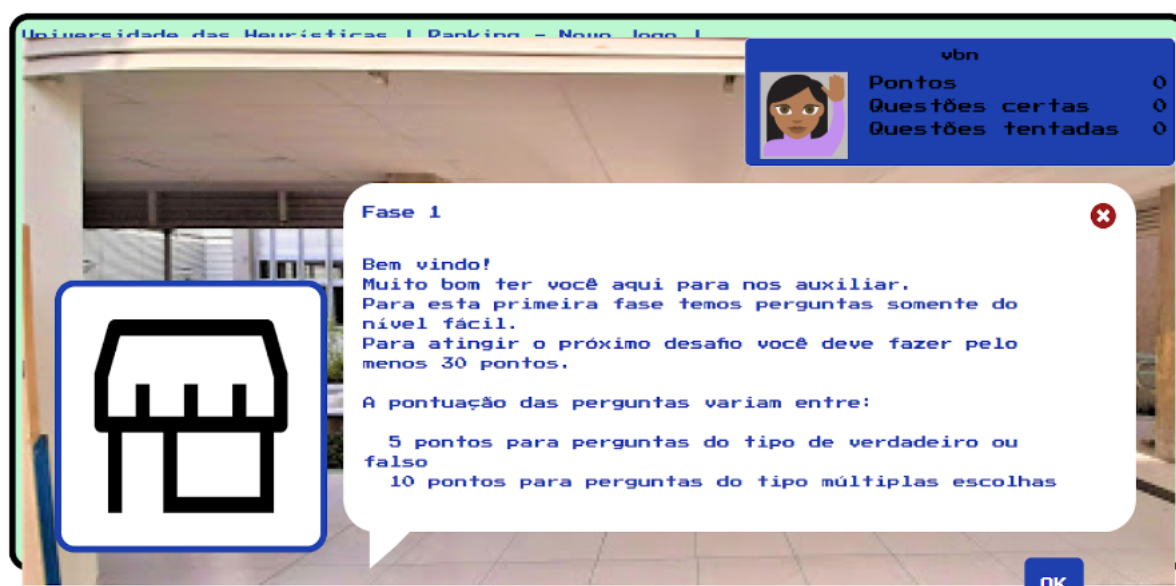
Fonte: Oliveira e Naves (2021)

Figura 38 – Tela cadastro usuário UnH.



Fonte: Oliveira e Naves (2021)

Figura 39 – Instruções para questionário UnH.



Fonte: Oliveira e Naves (2021)