

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

**Universidade das Heurísticas: Um jogo para o
apoio a aprendizagem das heurísticas de
Nielsen**

**Autores: Cauê Mateus Oliveira
Lucas Midlhey Cardoso Naves
Orientadora: Dra. Fabiana Freitas Mendes**

**Brasília, DF
2022**



Cauê Mateus Oliveira
Lucas Midlhey Cardoso Naves

Universidade das Heurísticas: Um jogo para o apoio a aprendizagem das heurísticas de Nielsen

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

Orientadora: Dra. Fabiana Freitas Mendes

Coorientador: Dr. André Barros de Sales

Brasília, DF

2022

Cauê Mateus Oliveira

Lucas Midlhey Cardoso Naves

Universidade das Heurísticas: Um jogo para o apoio a aprendizagem das heurísticas de Nielsen/ Cauê Mateus Oliveira

Lucas Midlhey Cardoso Naves. – Brasília, DF, 2022-

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

Orientadora: Dra. Fabiana Freitas Mendes

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB

Faculdade UnB Gama – FGA , 2022.

1. Heurísticas de Nielsen. 2. Jogo ensino-aprendizagem. I. Dra. Fabiana Freitas Mendes. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Universidade das Heurísticas: Um jogo para o apoio a aprendizagem das heurísticas de Nielsen

CDU

Cauê Mateus Oliveira
Lucas Midlhey Cardoso Naves

Universidade das Heurísticas: Um jogo para o apoio a aprendizagem das heurísticas de Nielsen

Monografia submetida ao curso de graduação
em Engenharia de Software da Universidade
de Brasília, como requisito parcial para ob-
tenção do Título de Bacharel em Engenharia
de Software.

Trabalho aprovado. Brasília, DF, 09 de junho de 2022.

Dra. Fabiana Freitas Mendes
Orientadora

Dr. André Barros de Sales
Coorientador

Dr. André Luiz Peron Martins Lanna
Convidado

**Msc. Eduardo Gabriel Queiroz
Palmeira**
Convidado

Brasília, DF
2022

**Este trabalho é dedicado aos futuros estudantes de cursos da área de
Ciências da Computação.**

Agradecimentos

Agradecemos à nossa orientadora Fabiana Freitas Mendes e nosso coorientador André Barros de Sales por aceitar conduzir nosso trabalho de pesquisa.

A todos os nossos professores do curso de Engenharia de *Software* da Universidade de Brasília pela excelência da qualidade técnica de cada um.

Somos gratos às nossas famílias pelo apoio que sempre nos deram durante toda a nossa vida.

Aos nossos pais Fábio de Souza Oliveira e Leonardo Cardoso Naves, e as nossas mães Rosilene Mateus Oliveira e Flávia Midlhey Grillo que sempre estiveram ao nosso lado nos apoiando ao longo de toda a nossa trajetória.

Às nossas namoradas Joyce Feitosa da Silva e Emanuela dos Santos Meireles pela compreensão e paciência demonstrada durante o período do projeto.

“A tarefa não é tanto ver aquilo que ninguém viu, mas pensar o que ninguém ainda pensou sobre aquilo que todo mundo vê.” (Arthur Schopenhauer)

Resumo

O ensino de Interação Humano-Computador (IHC) provê ferramentas para que o profissional de Engenharia de *Software* entenda a forma que o usuário se comunica com a máquina. Os avanços da tecnologia têm permitido a adaptação do ensino na área de *software* influenciando os estudos e tornando-os mais interativos. Os jogos educacionais auxiliam o processo de ensino-aprendizagem como um complemento às aulas. Deste modo, o objetivo deste trabalho é desenvolver um jogo com foco no processo de ensino-aprendizagem que englobe o conhecimento de heurísticas de Nielsen no contexto de IHC e possibilite ao jogador aprender mais facilmente o conteúdo. Inicialmente foi conduzida uma revisão bibliográfica e, depois disso, o restante do trabalho foi dividido em quatro fases: (1) elaboração do projeto de pesquisa; (2) coleta, organização e análise de material; (3) desenvolvimento do jogo; e (4) coleta, organização e análise de dados. Os resultados deste trabalho são os requisitos elicitados do jogo Universidade das Heurísticas, os protótipos de baixa e alta fidelidade utilizados para validação da ideia, modelo, *design* do jogo, arquitetura do jogo e o próprio jogo desenvolvido.

Palavras-chave: Heurísticas de Nielsen; Usabilidade; Jogo Educacional; Ensino-Aprendizagem.

Abstract

The Human-Computer Interaction (HCI) provides tools to Software engineers for understanding how user communicates with machine. Advances in technology have allowed teaching adaptations in the software area, influencing studies and making them more interactive. Educational games support the teaching-learning process as a complement to classes. Thereby, the goal of this work is to develop a game focused on the teaching-learning process that encompasses knowledge of Nielsen's heuristics in the context of HCI and allows the player to learn the content more easily. Initially a bibliographic review was conducted, and after that, the remaining work was divided into four phases: (1) elaboration of the research project; (2) collection, organization and analysis of material; (3) game development; (4) and data collection, organization and analysis. The results of this work are the elicited requirements of the game University of Heuristics, the low and high fidelity prototypes used to validate the idea, the model, design of the game, game architecture and the game developed.

Key-words: Nielsen Heuristics; Usability; Educacional Game; Teaching-Learning.

Lista de ilustrações

Figura 1 – EAP	20
Figura 2 – Fases da pesquisa	25
Figura 3 – Etapas da fase de elaboração do projeto de pesquisa	26
Figura 4 – Fases do <i>Playcentric Design Process</i>	27
Figura 5 – Eventos do Scrum	28
Figura 6 – Etapas da fase de coleta, organização e análise de material	29
Figura 7 – Etapas da fase de desenvolvimento do jogo	30
Figura 8 – Etapas da fase de coleta, organização e análise de dados	31
Figura 9 – Processo de iteração	34
Figura 10 – Painel genérico do Kanban	38
Figura 11 – Terminologias dos <i>Serious Games</i> (visão geral dos termos básicos)	42
Figura 12 – Imagem de seleção de personagem	48
Figura 13 – Exemplo de pergunta do tipo múltipla escolha	50
Figura 14 – Exemplo de pergunta do tipo verdadeiro ou falso	50
Figura 15 – Representação arquitetural do Jogo	57
Figura 16 – Representação da Integração contínua e Implantação contínua	58
Figura 17 – Diagrama de Classes	61
Figura 18 – Diagrama de atividades	62
Figura 19 – Diagrama de pacotes representação <i>front-end</i>	63
Figura 20 – Diagrama de pacotes representação <i>back-end</i>	64
Figura 21 – Tela Inicial do jogo	65
Figura 22 – Menu Inicial do jogo	66
Figura 23 – Tela de boas vindas do jogo	66
Figura 24 – Tela das fases do jogo	67
Figura 25 – Descrição da primeira fase	68
Figura 26 – Pergunta de múltipla escolha	68
Figura 27 – Pergunta de verdadeiro ou falso	69
Figura 28 – <i>Feedback</i> resposta correta	69
Figura 29 – <i>Feedback</i> resposta incorreta	70
Figura 30 – Tela de <i>Ranking</i>	70
Figura 31 – Tela do jogo com informações adicionais	71
Figura 32 – Tela inicial do jogo	83
Figura 33 – Tutorial inicial do jogo	84
Figura 34 – Escolha dos níveis para se jogar	84
Figura 35 – Início do jogo com explicação do nível um	85
Figura 36 – Início do nível um com a primeira pergunta	85

Figura 37 – Feedback quando o usuário errar a pergunta	86
Figura 38 – Segunda pergunta do jogo mostrando que dentro dos níveis terão perguntas binárias e de múltipla escolha	86
Figura 39 – Mensagem quando o usuário acertar a pergunta	87
Figura 40 – Sequência de perguntas do jogo	87
Figura 41 – Mensagem de acerto do usuário	88
Figura 42 – Feedback ao final do nível 1 dizendo se o usuário atingiu o necessário para passar de nível ou não	88
Figura 43 – Ranking do jogo	89
Figura 44 – Página inicial do jogo	90
Figura 45 – Menu inicial para acessar ranking e jogar	90
Figura 46 – Caixa de diálogo que é mostrado após clicar em jogar	91
Figura 47 – Introdução ao jogo e uma breve história	91
Figura 48 – Apresentação da tela do jogo com a disposição dos projetos	92
Figura 49 – Breve introdução do primeiro projeto	92
Figura 50 – Feedback de pontuação quando acerta a resposta	93
Figura 51 – Segunda pergunta a ser respondida de verdadeiro ou falso	94
Figura 52 – Feedback quando erra a questão	94
Figura 53 – Página de ranking acessada ao menu principal	95
Figura 54 – Pergunta do nível um de múltipla escolha	98
Figura 55 – Pergunta do nível um de múltipla escolha	99
Figura 56 – Pergunta do nível um de múltipla escolha	99
Figura 57 – Pergunta do nível um de múltipla escolha	100
Figura 58 – Pergunta do nível um de múltipla escolha	101
Figura 59 – Pergunta do nível um de múltipla escolha	102
Figura 60 – Pergunta do nível um de múltipla escolha	103
Figura 61 – Pergunta do nível um de múltipla escolha	104
Figura 62 – Pergunta do nível um de múltipla escolha	105
Figura 63 – Pergunta do nível um de múltipla escolha	106
Figura 64 – Perguntas de nível um de verdadeiro ou falso	107
Figura 65 – Perguntas do nível um de verdadeiro ou falso	108
Figura 66 – Perguntas do nível um de verdadeiro ou falso	109
Figura 67 – Pergunta do nível dois de múltipla escolha	110
Figura 68 – Pergunta do nível dois de múltipla escolha	111
Figura 69 – Pergunta do nível dois de múltipla escolha	112
Figura 70 – Pergunta do nível dois de múltipla escolha	113
Figura 71 – Pergunta do nível dois de múltipla escolha	114
Figura 72 – Pergunta do nível dois de múltipla escolha	115
Figura 73 – Pergunta do nível dois de múltipla escolha	116

Figura 74 – Pergunta do nível dois de múltipla escolha	117
Figura 75 – Pergunta do nível dois de múltipla escolha	118
Figura 76 – Pergunta do nível dois de múltipla escolha	119
Figura 77 – Pergunta do nível dois de verdadeiro ou falso	120
Figura 78 – Pergunta do nível dois de verdadeiro ou falso	121
Figura 79 – Pergunta do nível dois de verdadeiro ou falso	122
Figura 80 – Pergunta do nível três de múltipla escolha	123
Figura 81 – Pergunta do nível três de múltipla escolha	124
Figura 82 – Pergunta do nível três de múltipla escolha	124
Figura 83 – Pergunta do nível três de múltipla escolha	125
Figura 84 – Pergunta do nível três de múltipla escolha	126
Figura 85 – Pergunta do nível três de múltipla escolha	126
Figura 86 – Pergunta do nível três de múltipla escolha	127
Figura 87 – Pergunta do nível três de múltipla escolha	128
Figura 88 – Pergunta do nível três de múltipla escolha	129
Figura 89 – Pergunta do nível três de múltipla escolha	130
Figura 90 – Pergunta do nível três de verdadeiro ou falso	131
Figura 91 – Pergunta do nível três de verdadeiro ou falso	132
Figura 92 – Pergunta do nível três de verdadeiro ou falso	133
Figura 93 – Persona primária	136
Figura 94 – Persona secundária	137
Figura 95 – Persona suplementar	138
Figura 96 – Anti persona	139

Lista de tabelas

Tabela 1 – Pontos de melhorias identificados nos jogos <i>UsabiliCity</i> , <i>UsabilityGame</i> e <i>MACteaching</i>	18
Tabela 2 – Cronograma do TCC 1	21
Tabela 3 – Cronograma do TCC2	22
Tabela 4 – Problemas e sugestões de correção da validação do protótipo de baixa fidelidade	53
Tabela 5 – Avaliação do protótipo de baixa fidelidade	53
Tabela 6 – Problemas e sugestões de correção da validação do protótipo de alta fidelidade	55
Tabela 7 – Avaliação do protótipo de alta fidelidade	55
Tabela 8 – Tecnologias escolhidas e sua justificativa	56
Tabela 9 – Requisitos Funcionais	59
Tabela 10 – Requisitos Não-Funcionais	60
Tabela 11 – Vantagens e desvantagens das ferramentas de Virtualização do sistema	80
Tabela 12 – Vantagens e desvantagens para escolha do banco de dados	80
Tabela 13 – Vantagens e desvantagens das ferramentas de desenvolvimento do <i>front-end</i>	81
Tabela 14 – Vantagens e desvantagens das ferramentas de desenvolvimento do <i>back-end</i>	81
Tabela 15 – Pontos fortes e fracos e justificativa de escolha da ferramenta de Integração e implantação contínua	82
Tabela 16 – Tabela com perguntas, níveis e referências do banco de dados	97
Tabela 17 – Itens de instrumento de medição MEEGA+	135

Lista de abreviaturas e siglas

DAX	Expressões de análise de dados
DOM	Modelo de objeto de documento
EAD	Ensino A Distância
EAP	Estrutura Analítica do Projeto
GDD	Documento de projeto do jogo
IHC	Interação Humano-Computador
ISO	Organização Internacional para Padronização
IU	Interface de Usuário
MAC	Método de Avaliação de Comunicabilidade
MEGAA+	Modelo de Avaliação de Jogos Educacionais para o Ensino de Computação
MVC	<i>Model-View-Controller</i>
TCC	Trabalho de Conclusão de Curso
TDD	Documento técnico de concepção
TI	Tecnologia da Informação
UnB	Universidade de Brasília
UX	Experiência de usuário
VBA	Virtual Basic para aplicações
XP	<i>eXtreme Programming</i>

Sumário

1	INTRODUÇÃO	17
1.1	Justificativa	17
1.2	Objetivos	18
1.2.1	Objetivo Geral	19
1.2.2	Objetivos Específicos	19
1.3	Definição do Escopo	19
1.4	Plano de Estudo TCC1 e TCC2	21
1.5	Estrutura do Trabalho	23
2	METODOLOGIA	24
2.1	Elaboração do projeto de pesquisa	25
2.1.1	Definição da Metodologia	26
2.1.1.1	Passo a passo do Processo de concepção centrado no jogo (<i>Playcentric Design Process</i>)	27
2.1.1.2	Passo a passo do <i>Scrum</i> e <i>Kaban</i>	28
2.2	Coleta, organização e análise de material	29
2.3	Desenvolvimento do jogo	30
2.4	Coleta, organização e análise de dados	31
3	REFERENCIAL TEÓRICO	33
3.1	Processo de desenvolvimento de jogo	33
3.1.1	Processo de design centrado no jogador (<i>Playcentric Design Process</i>)	33
3.2	Processo de desenvolvimento de software	35
3.2.1	<i>Scrum</i>	35
3.2.2	<i>Kanban</i>	37
3.3	Interação humano-computador	38
3.4	Heurísticas de Usabilidade de Nielsen	40
3.4.1	Contexto de aplicação	40
3.4.2	Definição das heurísticas de Nielsen de interface	40
3.4.3	Avaliação heurística de interface	41
3.5	Jogos para ensino	42
3.6	Trabalhos correlatos	43
3.7	Tecnologia	44
4	CONCEPÇÃO DO JOGO	46
4.1	Ciclo de vida do jogo	46

4.2	Documento de projeto do jogo (<i>Game design document</i>)	46
4.2.1	Características gerais	46
4.2.2	Jogabilidade (<i>Gameplay</i>)	47
4.2.3	Elementos do jogo	48
4.2.3.1	Personagem	48
4.2.3.2	Perguntas	48
4.2.3.3	<i>Ranking</i>	51
4.2.4	Protótipo de baixa fidelidade	51
4.2.4.1	Processo de validação do protótipo de baixa fidelidade	51
4.2.5	Protótipo de alta fidelidade	53
4.2.5.1	Avaliação do protótipo de alta fidelidade	54
4.3	Documento técnico de concepção (<i>Technical design document</i>)	55
4.3.1	Tecnologias escolhidas	56
4.3.2	Arquitetura	57
4.3.3	Integração contínua e Implantação contínua	58
4.3.4	Requisitos	59
4.3.5	Diagramas	60
4.3.5.1	Diagrama de Classes	60
4.3.5.2	Diagrama de Atividades	61
4.3.5.3	Diagrama de Pacotes	63
5	UNIVERSIDADE DAS HEURÍSTICAS	65
6	CONCLUSÃO	72
	REFERÊNCIAS	74
	APÊNDICES	79
	APÊNDICE A – ESCOLHA DE TECNOLOGIAS	80
	APÊNDICE B – PROTÓTIPO DE BAIXA FIDELIDADE	83
	APÊNDICE C – PROTÓTIPO DE ALTA FIDELIDADE	90
	APÊNDICE D – BANCO DE DADOS DAS PERGUNTAS	96
D.1	Perguntas do nível um	98
D.1.1	Perguntas de múltipla escolha	98
D.1.2	Perguntas de verdadeiro ou falso	107
D.2	Perguntas do nível dois	110

D.2.1	Perguntas de múltipla escolha	110
D.2.2	Perguntas de verdadeiro ou falso	120
D.3	Perguntas do nível três	123
D.3.1	Perguntas de múltipla escolha	123
D.3.2	Perguntas de verdadeiro ou falso	131

ANEXOS **134**

ANEXO A – MODELO MEEGA+ 135

ANEXO B – PERSONAS UTILIZADAS NO PROJETO 136

1 Introdução

A interação entre humanos e computadores tem crescido a cada década, em virtude da evolução das tecnologias. Essa relação torna o comportamento humano mais dependente das novas ferramentas, mas por outro lado, os autores [Barbosa et al. \(2021\)](#) apontam que a produtividade aumentou de modo considerável nos últimos anos, devido a automação e agilidade das máquinas.

Os estudos desenvolvidos na área de Interação Humano-Computador (IHC) são aplicados em diversos sistemas, especialmente dentro do contexto da Engenharia de *Software*. Segundo [Hewett et al. \(1992\)](#), a área de IHC tem como foco a qualidade de uso de sistema interativos e busca trabalhar os *softwares* de “fora para dentro”, ou seja, é direcionado primeiro para os usuários e depois para o código. Tal justificativa reforça a importância do estudo de conteúdos abordados em IHC pelos profissionais da área de tecnologia da informação (TI) ([PEREIRA, 2013](#)). Nesse contexto, Nielsen é o autor que possui alguns dos mais consolidados princípios e diretrizes na construção de interfaces de usuário, conhecidas como heurísticas de Nielsen ([BARBOSA et al., 2021](#)).

A IHC tem se adaptado com sucesso no Brasil e teve seu processo acelerado após o decreto da pandemia de Sars-Cov-2, pela Organização Mundial da Saúde. Uma das principais estratégias utilizadas consistem nos jogos digitais educacionais ([MARQUES; AMARAL, 2020](#)). Estes, possuem grande importância na educação dos alunos, pois são fonte de motivação, diversão e aprendizagem, de modo simultâneo ([TAROUÇO et al., 2004](#)).

Os jogos são capazes de estimular o sistema neurológico, resolver problemas e alcançar conquistas que promovem o bem-estar do usuário ([TOLOMEI, 2017](#)). Neste contexto, a produção deste trabalho objetivou demonstrar um processo de desenvolvimento de um jogo para navegadores *web*, fundamentado nas heurísticas de Nielsen, com o intuito de auxiliar o ensino-aprendizagem de alunos de Engenharia de *Software*. Desenvolver um jogo que reforce o processo de ensino-aprendizagem na área de IHC auxilia a educação em sala de aula e a educação a distância (EAD), além de validar o aprendizado do aluno. A próxima seção descreve a justificativa deste trabalho.

1.1 Justificativa

Ao buscar na literatura, foram identificados jogos com ênfase em IHC e nas heurísticas de Nielsen, que auxiliam estudantes no processo de ensino-aprendizagem. Os resultados da pesquisa evidenciaram a necessidade de melhorias e adaptações nos jogos para

ampliar a satisfação e motivação de alunos da área de computação (CLÍMACO, 2016). Na Tabela 1 é possível visualizar estudos que analisaram jogos similares ao proposto no presente trabalho. As três pesquisas apresentaram oportunidades de melhoria nos jogos, tais como fases dinâmicas, seleção de personagens e explicação do jogo, que são utilizadas neste trabalho para aprimoramento das estratégias.

Tabela 1 – Pontos de melhorias identificados nos jogos *UsabiliCity*, *UsabilityGame* e *MACteaching*

Jogo	Melhorias	Referência
UsabiliCity	- Fases dinâmicas - Seleção de personagens	(FERREIRA et al., 2014)
UsabilityGame	- Explicação do jogo	(SOMMARIVA et al., 2011)
MACteaching	- Explicação do jogo	(BRITO et al., 2016)

Fonte: Elaboração própria (2021)

As oportunidades de melhoria identificadas nos jogos *UsabiliCity*, *UsabilityGame* e *MACteaching* foram implementadas no jogo desenvolvido neste trabalho. Como um diferencial aos três jogos pesquisados: as fases do jogo Universidade das Heurísticas não são estáticas, variando as perguntas apresentadas aos jogadores; no jogo Universidade das Heurísticas o jogador pode escolher qual personagem deseja usar; e as regras do jogo são apresentadas ao jogador.

A implementação das fases dinâmicas e seleção de personagens, em comparação com o *UsabiliCity*, acontecem da seguinte forma, respectivamente: perguntas em ordem aleatória e sem repetição constante e a opção do jogador poder escolher seu personagem e o nome do mesmo no início do jogo.

A implementação da explicação do jogo, em comparação com o *UsabilityGame* e o *MACteaching*, acontece da seguinte forma: antes do usuário começar a responder as perguntas, o jogador é inserido no contexto do jogo, e posteriormente, é explicado a ele as regras de cada nível. Na próxima seção são descritos os objetivos gerais e específicos deste trabalho.

1.2 Objetivos

Os objetivos são divididos em geral e específicos. O objetivo geral traz o propósito completo do trabalho e os específicos expõem a visão mais detalhada destes objetivos (GIL, 2002).

1.2.1 Objetivo Geral

Criar um jogo digital com foco no auxílio do ensino-aprendizagem de heurísticas de Nielsen.

1.2.2 Objetivos Específicos

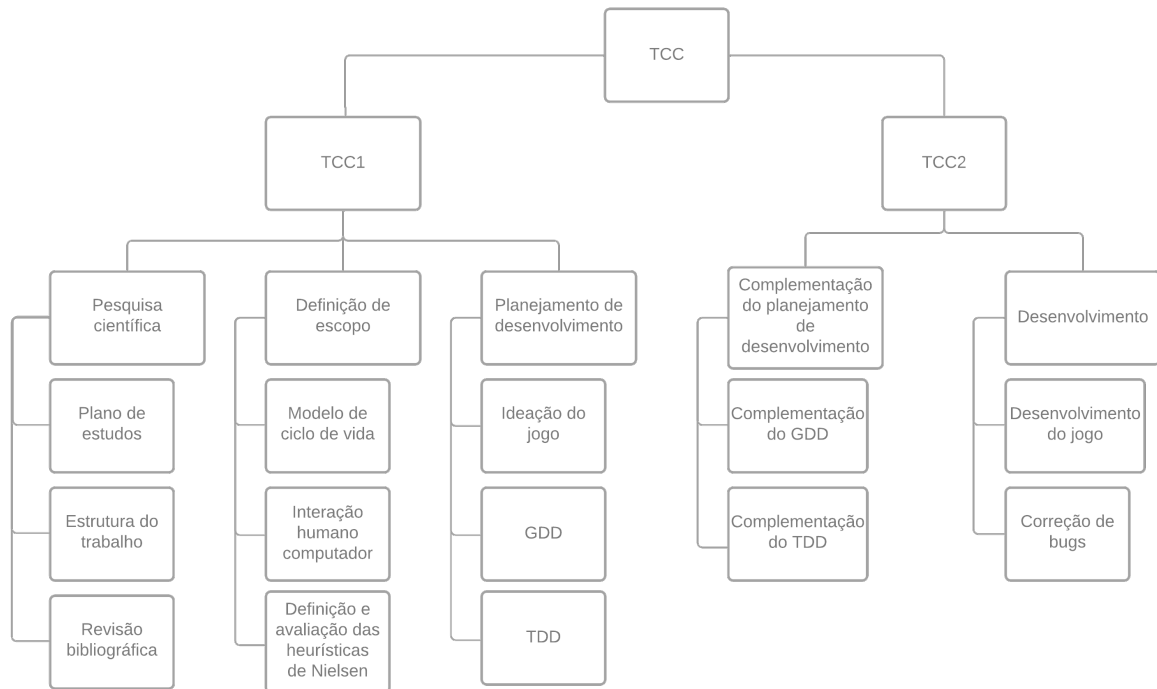
- **OE01** - Investigar os conceitos relacionados a jogos digitais de apoio ao processo de ensino-aprendizagem.
- **OE02** - Investigar as heurísticas de Nielsen.
- **OE03** - Desenvolver um jogo para ensino-aprendizagem.
- **OE04** - Utilizar boas práticas de programação e processo de desenvolvimento de *software* no desenvolvimento do jogo.
- **OE05** - Utilizar os conceitos das heurísticas de usabilidade no jogo.
- **OE06** - Desenvolver melhorias identificadas nos jogos *UsabiliCity*, *UsabilityGame* e *MACteaching*.

Na próxima seção estão descritas as etapas do trabalho, processo de pesquisa e estrutura analítica do projeto.

1.3 Definição do Escopo

Os *stakeholders* do projeto são estudantes de Engenharia de *Software* que buscam capacitação no tema das heurísticas de usabilidade de Nielsen (NIELSEN, 2005). Para a produção do jogo foi estabelecida uma Estrutura Analítica do Projeto (EAP) que está elencada em dois grandes blocos: TCC1 e TCC2. A EAP híbrida é composta por etapas e subetapas, conforme ilustrado na Figura 1.

Figura 1 – EAP



Fonte: Elaboração própria (2022)

A EAP foi dividida em subetapas, que foram desenvolvidas durante todo o ciclo de trabalho, com o propósito de alcançar e cumprir os objetivos específicos propostos (ALMEIDA, 2020). Os seis objetivos específicos do trabalho estão distribuídos entre as cinco etapas presentes na Figura 1: pesquisa científica, definição de escopo, planejamento e desenvolvimento, complementação do planejamento de desenvolvimento e desenvolvimento.

A etapa de avaliação não está presente na EAP, tendo sido somente planejada, mas não foi realizada, por falta de tempo hábil para a realização da coleta de dados para análise do aprendizado e desempenho dos jogadores. Cada etapa da EAP possui sub-tarefas descritas a seguir.

A etapa de **pesquisa científica** buscou revisar na literatura conceitos acerca de heurísticas de usabilidade e jogos digitais. Por meio da execução dessa etapa, atinge-se o objetivo OE01 (ver Seção 1.2.1). A etapa de **definição de escopo** consiste nos resultados encontrados pela busca de artigos científicos, modelo de ciclo de vida e as heurísticas de Nielsen. Essa etapa teve o intuito de cumprir o objetivo OE02 (ver Seção 1.2.1). A etapa de **planejamento e desenvolvimento** refere-se a aplicação de conceitos, documentos específicos de desenvolvimento e etapas práticas. A conclusão desta etapa garantiu que os objetivos OE03, OE04, OE05 e OE06 fossem alcançados (ver Seção 1.2.1).

A etapa de **complementação do planejamento de desenvolvimento** refere-se

a continuidade proposta no item anterior. Esta etapa concluída representou os objetivos OE3, OE4, OE5 e OE06 (ver Seção 1.2.1). A etapa de **desenvolvimento** refere-se a produção do jogo, correção de *bugs* e as boas práticas para desenvolvimento de *software*. Esta etapa concluída garantiu que os objetivos OE3, OE04, OE05 e OE06 fossem alcançados (ver Seção 1.2.1).

Na próxima seção está descrito o cronograma do trabalho em sua primeira e segunda fase.

1.4 Plano de Estudo TCC1 e TCC2

As atividades desenvolvidas na pesquisa estão previstas de acordo com dois cronogramas, que incluem a apresentação das etapas do TCC1 e TCC2, determinados pelo prazo de entrega e as atividades que antecedem as apresentações são subdivididas em *sprints* semanais. Na Tabela 2 é possível ter acesso ao cronograma de tarefas detalhado do TCC1, já na Tabela 3 está apresentado o cronograma do TCC2.

Tabela 2 – Cronograma do TCC 1

Data de inicio	Data de fim	Atividade
08/02/2021	14/02/2021	Definição do cronograma.
08/02/2021	21/02/2021	Estrutura do TCC.
08/02/2021	02/05/2021	Redação do TCC.
08/02/2021	02/05/2021	Revisão bibliográfica.
08/02/2021	28/02/2021	Definição do escopo, motivação e justificativa.
08/02/2021	28/02/2021	Trabalhos correlatos.
15/02/2021	28/02/2021	Definição do tema, objetivo geral e objetivo específico.
01/03/2021	14/03/2021	Processo de desenvolvimento do jogo - <i>Playcentric Design Process</i> .
01/03/2021	14/03/2021	Metodologia de software - Tecnologia e ciclo de vida.
01/03/2021	14/03/2021	Interação humano computador e heurística de Nielsen.
08/02/2021	02/05/2021	GDD.
08/02/2021	02/05/2021	TDD.
03/05/2021	09/05/2021	Revisão do TCC.
03/05/2021	09/05/2021	Entrega do TCC para banca.
10/05/2021	23/05/2021	Defesa de TCC1.

Fonte: Elaboração própria (2021)

As atividades apresentadas na Tabela 2 foram programadas para serem realizadas nas datas estipuladas dentro das 15 semanas. Neste período planejado todas as atividades conseguiram ser concluídas, porém com ajustes de execução nas datas das atividades entre as *sprints*.

Tabela 3 – Cronograma do TCC2

Data de início	Data de fim	Atividade
22/11/2021	28/11/2021	Validação do protótipo de alta fidelidade.
22/11/2021	28/11/2021	Correção da documentação.
29/11/2021	12/12/2021	Configuração do ambiente de desenvolvimento.
06/12/2021	02/01/2022	Diagrama de classe, atividade e pacotes.
06/12/2021	02/01/2022	Estudo das tecnologias.
27/12/2021	16/01/2022	Criação do <i>front-end</i> e estrutura do banco de dados.
17/01/2022	30/01/2022	Criação das <i>models</i> .
24/01/2022	30/01/2022	Complementação da escrita da arquitetura do jogo.
31/01/2022	20/03/2022	Implementação das três fases do jogo e dos níveis de dificuldade.
31/01/2022	20/03/2022	Evolução da escrita do GDD e TDD.
07/03/2022	03/04/2022	Implementação do <i>feedback</i> das respostas, pontuação do jogador e ranking.
04/04/2022	24/04/2022	Correções de <i>bugs</i> e ajustes na escrita.
25/04/2022	01/05/2022	Entrega do trabalho escrito e do jogo para banca e montagem da apresentação.
02/05/2022	08/05/2022	Apresentação final do trabalho.

Fonte: Elaboração própria (2022)

As atividades apresentadas na Tabela 3 foram programadas para serem realizadas nas datas estipuladas dentro das 21 semanas. Neste período planejado todas as atividades conseguiram ser concluídas, porém com ajustes de execução nas datas das atividades entre as *sprints*. O tempo para desenvolvimento das atividades no TCC2 acabou sendo maior, pois o início dos trabalhos nesta etapa acabaram se iniciando antes do semestre letivo se iniciar.

Percebe-se que entre o final do cronograma do TCC1 para o início do TCC2 possui um intervalo de tempo de um semestre entre as etapas do trabalho. Esse intervalo é

explicado devido a pausa que os autores fizeram entre a primeira e a segunda etapa do trabalho por motivos profissionais. Na próxima seção estão detalhadas cada capítulo do projeto e sua definição de pesquisa.

1.5 Estrutura do Trabalho

O trabalho está estruturado em cinco capítulos. O Capítulo 1 apresenta uma breve introdução sobre o tema, definições, justificativa de desenvolvimento, bem como os objetivos da pesquisa.

O Capítulo 2 refere-se a metodologia de pesquisa, materiais e métodos para desenvolvimento das etapas propostas. Por meio desta etapa foi possível descrever as tecnologias aplicadas, ciclos de vida, as heurísticas de Nielsen atreladas e a usabilidade no contexto de IHC.

No Capítulo 3 está disposto o referencial teórico, que embasa o processo de desenvolvimento do jogo através do modelo para avaliação de jogos educativos (*model for the evaluation of educational games - MEEGA+*), metodologias *Scrum* e *Kanban*. Nesse mesmo capítulo é apresentado um mapeamento de IHC, contexto de aplicação, definição e avaliação das heurísticas de Nielsen para o desenvolvimento do *software*.

O Capítulo 4 apresenta os resultados; nele foi feito o processo documental do documento técnico de concepção (*technical design document - TDD*) e documento de projeto do jogo (*game design document - GDD*), que representam, no jogo, as características gerais, análise de jogabilidade, elementos do jogo, protótipo de baixa e alta fidelidade, tecnologias escolhidas, arquitetura, levantamento de requisitos e diagramas.

O Capítulo 5 apresenta o jogo Universidade das Heurísticas por meio de imagens das telas do jogo já desenvolvido. É mostrado desde a tela inicial do jogo até a classificação (*ranking*) com as pontuações dos jogadores que jogaram o jogo até o final.

No Capítulo 6 está a conclusão do trabalho, onde é retomado todos os tópicos que foram apresentados, bem como os objetivos geral e específicos e a proposta do trabalho. É apresentado também as contribuições do trabalho para os autores e o contexto em que o jogo será utilizado.

2 Metodologia

Neste capítulo é apresentada a classificação e metodologia de pesquisa utilizada neste trabalho e as etapas propostas para cada fase da pesquisa. O detalhamento das fases pode ser verificado nas Seções de 2.1 a 2.4.

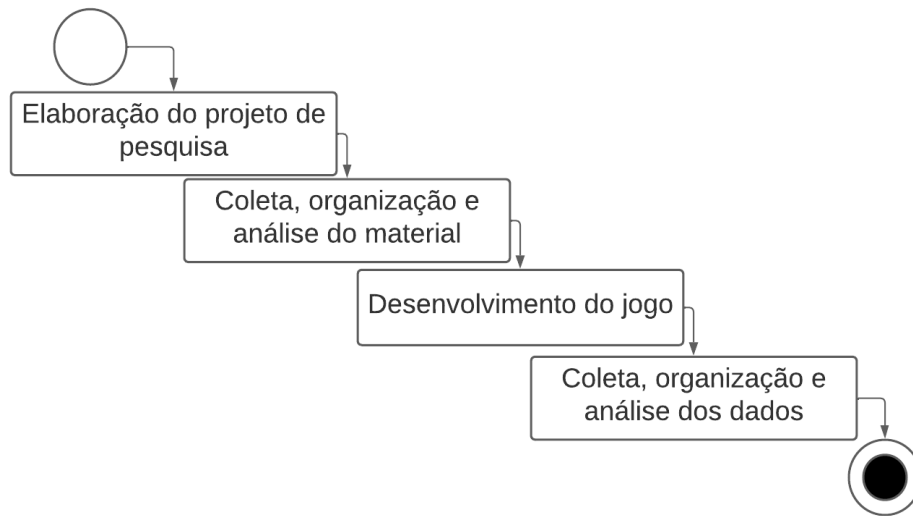
De acordo com Kauark et al. (2010), uma pesquisa pode ser classificada quanto a (1) natureza, (2) forma de abordagem, (3) objetivos, (4) procedimentos técnicos adotados e (5) entre fenomênica ou factual.

Este trabalho possui sua **natureza** em pesquisa aplicada, pois se trata do desenvolvimento de uma solução para um problema específico. A **forma de abordagem**, trata-se de uma pesquisa qualitativa, pois foi criado somente o jogo, sem uma avaliação posterior. No que se refere ao ponto entre **fenomênica x factual**, a pesquisa é factual, já que trata-se de um trabalho com foco em solucionar um problema e aprimorar barreiras, por meio da criação de um jogo e produzindo influência sobre o processo de ensino-aprendizagem de estudantes.

Quanto aos **objetivos**, a pesquisa é do tipo exploratória, pois envolveu o levantamento bibliográfico e, também, tornou o problema existente explícito, que pode contribuir para o aperfeiçoamento de jogos e solução de problemas. Por fim, tratando-se dos **procedimentos técnicos**, a pesquisa é bibliográfica, já que foi desenvolvida com base em livros e artigos publicados sobre os temas abordados.

A pesquisa foi dividida em quatro fases: (1) elaboração do projeto de pesquisa; (2) coleta, organização e análise de material; (3) desenvolvimento do jogo; (4) coleta, organização e análise de dados, conforme ilustra a Figura 2.

Figura 2 – Fases da pesquisa



Fonte: Elaboração própria (2021)

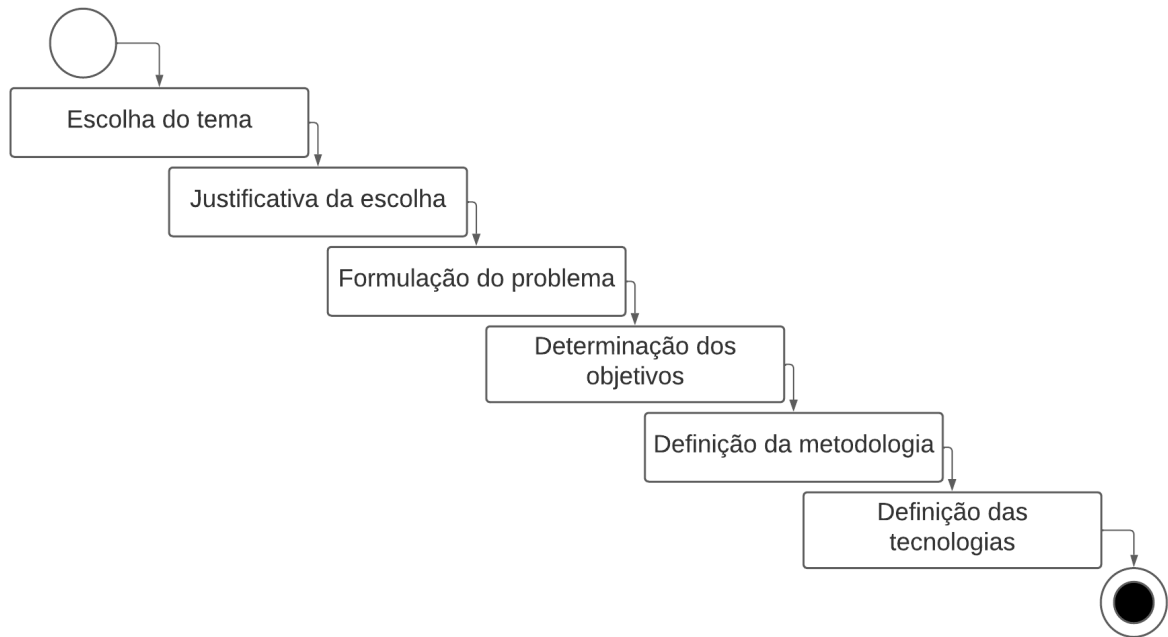
As fases da pesquisa apresentadas na Figura 2 foram apoiadas e adaptadas de Kauark et al. (2010). A fase de **elaboração do projeto de pesquisa** teve como objetivo identificar na literatura a importância de desenvolver a pesquisa, de modo a orientar a condução do trabalho. Já na fase de **coleta, organização e análise de material**, identificou-se os materiais que baseiam o desenvolvimento do jogo e auxiliam no levantamento de requisitos.

Na fase de **desenvolvimento do jogo** foi definido o protótipo para validar seu desenvolvimento. E, após esta etapa, além da efetuação do desenvolvimento, realizaram-se testes das funcionalidades do jogo para colocá-lo em produção. A última fase, denominada **pesquisa de coleta, organização e análise de dados**, não foi realizada, por falta de tempo hábil para a realização da coleta de dados para análise do aprendizado e desempenho dos jogadores. As próximas seções detalham cada uma das fases de pesquisa.

2.1 Elaboração do projeto de pesquisa

Nesta seção é apresentada as etapas da fase de elaboração do projeto de pesquisa. As etapas desta fase de pesquisa são ilustradas na Figura 3.

Figura 3 – Etapas da fase de elaboração do projeto de pesquisa



Fonte: Elaboração própria (2021)

Na primeira etapa, mostrada na Figura 3, foi definido o tema a partir de uma oportunidade de pesquisa na área de ensino-aprendizagem por meio da construção de um jogo sobre as heurísticas de usabilidade de Nielsen. Pesquisas semelhantes sobre as heurísticas embasam a justificativa na Seção 1.1.

A formulação do problema iniciou-se por meio da pesquisa de trabalhos correlatos. Posteriormente, planejou-se o desenvolvimento e estrutura do trabalho (Seção 1.4). Os objetivos do projeto foram definidos a partir dos problemas encontrados na literatura, como oportunidades para aprimorar as pesquisas existentes.

As tecnologias utilizadas no projeto foram definidas após a etapa da metodologia de desenvolvimento do jogo, por meio de um *brainstorm* entre os envolvidos na execução do trabalho. O detalhamento pode ser consultado no Apêndice A.

2.1.1 Definição da Metodologia

Nesta subseção são apresentadas as metodologias de desenvolvimento de jogo e de *software* utilizadas no trabalho. As metodologias estão descritas nas seções 2.1.1.1 e 2.1.1.2.

Para apoiar o estudo e complementar o processo de desenvolvimento do jogo, algumas práticas de Engenharia de *Software* foram adotadas. Foram adaptados os *frameworks Scrum* e *Kanban* para o projeto, pois são práticas da metodologia ágil e se assemelham a metodologia do *Playcentric Design Process*, que será abordada no capítulo subsequente.

A definição do projeto se estende a processos de desenvolvimento, tais como requisitos, análise, implementação, testes, implantação e qualidade.

2.1.1.1 Passo a passo do Processo de concepção centrado no jogo (*Playcentric Design Process*)

Segundo o *Playcentric Design Process* (FULLERTON, 2014), o processo de desenvolvimento do jogo é disposto em fases; a concepção, pré-produção, produção e garantia da qualidade do jogo está detalhado na Seção 3.1.1.

Foram realizadas adaptações na documentação e planejamento de validação para este projeto devido ao tempo, tamanho da equipe e finalidade do projeto. Na Figura 4 estão descritas as quatro fases seguidas pelo *Playcentric Design Process*.

Figura 4 – Fases do *Playcentric Design Process*



Fonte: Adaptado de Fullerton (2014)

Na Figura 4 foram apresentadas as quatro fases que foram utilizadas na metodologia de desenvolvimento, sendo elas a de concepção, pré-produção, produção, e garantia e qualidade. Cada fase possui etapas para concretização da mesma.

1. **Fase de concepção:** Nesta fase foi feito um *brainstorm*, referente a proposta do jogo, que representa a etapa de gerar ideias. Para validá-la, foram analisados os objetivos gerais e específicos do trabalho que representa a etapa de formalizar ideias.

2. **Fase de pré-produção:** Após validar a ideia, iniciou-se a etapa de teste; nesta etapa foi planejada a criação de protótipos que buscaram contextualizar o jogo e, conseqüentemente, permitiram melhorias da ideia formalizada seguindo o processo de iteração. A próxima etapa foi a de avaliar os resultados e revisá-los, para isso foram criados os artefatos GDD, TDD, melhores descritos nas seções 4.2 e 4.3, que puderam aproximar as ideias ao desenvolvimento.
3. **Fase de Produção:** Neste momento foram refinados os artefatos da fase de pré-produção, que podem ser vistos nas seções 4.2 e 4.3, bem como estética, história, mecânica e desenvolvimento do jogo. Assim como no final da fase de pré-produção, para cada etapa de avaliação dos resultados foi necessário revisar, em seguida testar, para posteriormente voltar a avaliar. Este ciclo foi finalizado assim que o jogo vai para produção e teste dos jogadores.
4. **Fase de Garantia e qualidade:** Nesta fase, o jogo seria disponibilizado ao público alvo para testes, na busca de retorno de possíveis melhorias (*feedbacks*). Nesta fase, o modelo cíclico representado na fase de produção continua até a sua data de lançamento. Para a produção deste jogo não foi alcançado esta etapa devido ao tempo hábil do trabalho.

2.1.1.2 Passo a passo do *Scrum* e *Kaban*

A utilização da ferramenta *Kanban* refere-se a organização e gestão de tarefas. Foram separados quatro cartões para este projeto: *To do* (A fazer), *Doing* (Fazendo), *Review* (Revisão), *Done* (Feito). Para mais informações sobre o *Kanban*, ver Seção 3.2.2.

O *Scrum* foi o *framework* utilizado para o desenvolvimento ágil deste trabalho. Foram definidas entregas semanais referentes a evolução do trabalho, planejadas reuniões diárias para alinhamento do desenvolvimento e reuniões semanais que validam a *Sprint*, conforme a Figura 5. Para mais informações sobre o *Scrum*, consultar Seção 3.2.1.

Figura 5 – Eventos do Scrum



Fonte: Elaboração própria (2021)

O **backlog do produto** representa a lista de tarefas pré-definidas pela equipe que foram concluídas ao final do trabalho. A **Sprint backlog** representa a lista de tarefas que foram feitas durante o período da *Sprint*. A definição das tarefas aconteceu no início de cada *Sprint*. A **Sprint** representa a etapa de execução do trabalho. O período durante a *Sprint* é de 7 dias corridos.

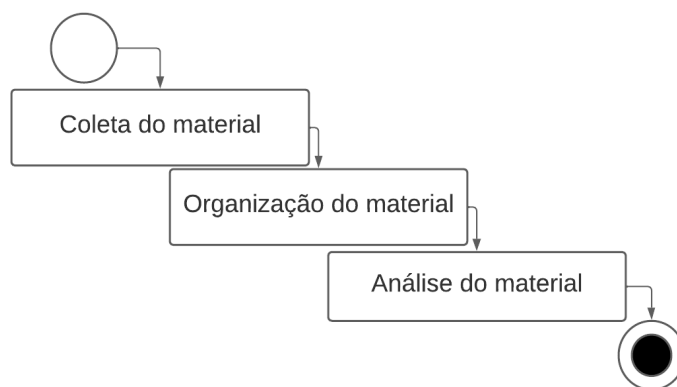
A **review** representa a etapa de validação do que foi executado durante a *Sprint*. A *review* acontece no primeiro dia da *Sprint* subsequente. O **done** representa o final da execução de todas as tarefas propostas no *backlog* do produto. Os cinco eventos do *Scrum* mostrados na Figura 5, foram detalhados para melhor entendimento do que se trata cada um deles.

Na próxima seção estão descritos as etapas de coleta, organização e análise do material utilizados para metodologia deste projeto.

2.2 Coleta, organização e análise de material

Nesta seção são apresentadas as etapas presentes dentro da fase de coleta, organização e análise de material. As etapas desta fase de pesquisa são ilustradas na Figura 6.

Figura 6 – Etapas da fase de coleta, organização e análise de material



Fonte: Elaboração própria (2021)

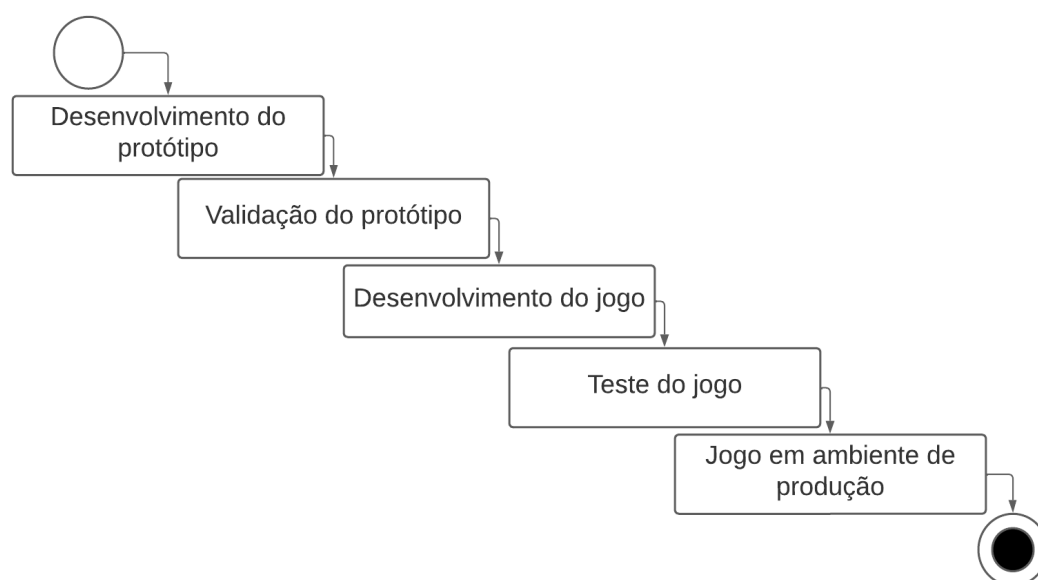
A busca pelos materiais de estudo, apresentada na Figura 6, foi realizada na ferramenta *Google Scholar*, que consiste em um mecanismo virtual de pesquisas acadêmicas e possui um acervo livros e artigos científicos. Segundo Harzing e Alakangas (2016), a ferramenta *Google Scholar* pode ser usada para realizar pesquisas científicas, já que se trata de um pesquisador de materiais livres e gratuitos para leitura. No Capítulo 3, os resultados da revisão bibliográfica podem ser encontrados nas seções com os temas relacionados a esse trabalho.

Os materiais foram filtrados obedecendo critérios de exclusão como: trabalhos contendo os temas de interesse, citados por outros autores e resumos pertinentes para o trabalho. Após a seleção dos artigos, foram organizados na ferramenta Zotero¹ para facilitar o armazenamento e o gerenciamento dos livros e artigos. A análise dos materiais foi aprofundada após a primeira etapa. Esses materiais selecionados constituem as referências para a escrita do trabalho e desenvolvimento do jogo. Na próxima subseção está descrita a fase de desenvolvimento do jogo, bem como seu planejamento.

2.3 Desenvolvimento do jogo

Nesta seção são apresentadas as etapas de desenvolvimento do jogo. As etapas desta fase são ilustradas na Figura 7.

Figura 7 – Etapas da fase de desenvolvimento do jogo



Fonte: Elaboração própria (2021)

Como mostrado na Figura 7, foi iniciada a fase de desenvolvimento do jogo após a fase de coleta, organização e análise dos materiais. Esta fase segue o modelo de iteração, que será descrito na Seção 3.1.1, e o desenvolvimento dos protótipos, que foram validados e testados utilizando a metodologia ágil. Foi desenvolvido um protótipo de baixa fidelidade no papel e, após concluído o ciclo de iterações, tendo sido aprovado, um protótipo de alta fidelidade digital foi desenvolvido e seguido o mesmo processo de iteração do protótipo em papel. Os protótipos de baixa e alta fidelidade estão disponíveis nas Seções 4.2.4 e 4.2.5, respectivamente.

¹ <<https://www.zotero.org/>>

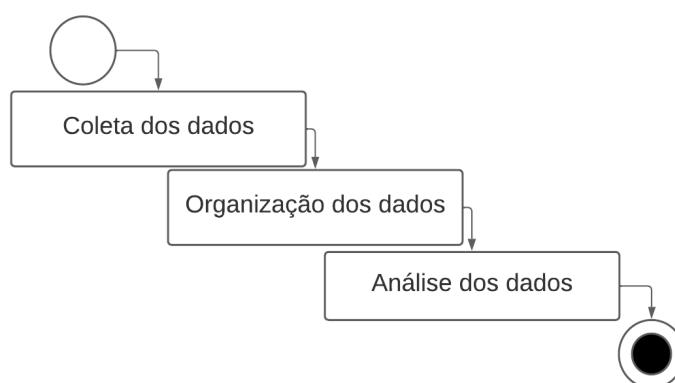
Os artefatos de GDD e TDD estão presentes na fase de desenvolvimento de jogo. O GDD documenta detalhes de como o jogo será: conceito, *design*, cronograma, plano de testes e mitigação de riscos. O TDD apresenta uma documentação técnica e detalhada sobre o jogo. Os principais elementos são: lista de funcionalidades, diagramas, lógica, conexões e *hardware/software* descritos no documento. Os artefatos encontram-se nas respectivas Seções 4.2 e 4.3.

Após o registro e refinamento dos artefatos, iniciou-se a etapa de programação do jogo. O jogo foi testado e validado seguindo a etapa de iteração, acoplada a metodologia *Scrum*. A etapa de teste do jogo foi realizada em conjunto ao desenvolvimento do jogo seguindo o modelo do *Playcentric Design Process*. O público-alvo foi definido a partir dos objetivos do trabalho, sendo o jogo voltado para alunos de graduação e pós-graduação de cursos da área de ciência da computação. Na próxima seção são apresentadas as etapas de coleta, organização e análise dos dados utilizadas para este projeto.

2.4 Coleta, organização e análise de dados

Nesta seção são apresentadas as etapas de coleta, organização e análise dos dados. Esta etapa foi planejada, mas não foi realizada, por falta de tempo hábil para a realização da coleta de dados para análise do aprendizado e desempenho dos jogadores. Nesta seção está descrito como foi planejado esta fase. As etapas desta fase são ilustradas na Figura 8.

Figura 8 – Etapas da fase de coleta, organização e análise de dados



Fonte: Elaboração própria (2021)

Como mostrado na Figura 8, após colocado o jogo em ambiente de produção (fase anterior), seria iniciada a coleta dos dados do jogo. Nesta fase, seria elaborado um questionário para os usuários do jogo responderem. As informações de uso do jogo seriam coletadas a partir de um monitoramento via sessão, que consiste em coletar informações de tempo da sessão, tempo para cada resposta e tempo que o jogador passa em cada opção

do menu. O questionário seria desenvolvido com o intuito de analisar as informações sobre qualidade do *software*, interesse dos usuários e se o jogo atende os objetivos definidos no projeto. A informação de monitoramento do usuário captaria o tempo do usuário por sessão e por fase do jogo.

A organização dos dados seria iniciada após a etapa de coleta, de acordo com o *feedback* dos usuários. Posteriormente, na etapa de análise seria observado as informações coletadas, comparadas e construídas considerações sobre as respostas. A análise subsidiaria a compreensão sobre a importância do jogo, especialmente se a criação de fato auxilia os estudantes no processo de ensino-aprendizagem. Esta etapa também buscava alinhar melhorias para o jogo. No próximo capítulo encontra-se o material utilizado como referência de pesquisa para este projeto. Este embasamento auxilia as etapas de desenvolvimento.

3 Referencial Teórico

Neste capítulo será apresentado o referencial teórico dos conhecimentos utilizados no projeto. O embasamento possibilitou o desenvolvimento do problema abordado no trabalho. Nas seções seguintes serão apresentados os seguintes temas: processo de desenvolvimento de jogo na Seção 3.1, processo de desenvolvimento de *software* na Seção 3.2, interação humano-computador na Seção 3.3, heurísticas de usabilidade de Nielsen na Seção 3.4, jogos para ensino na Seção 3.5, trabalhos correlatos na Seção 3.6 e, por fim, as tecnologias na Seção 3.7.

3.1 Processo de desenvolvimento de jogo

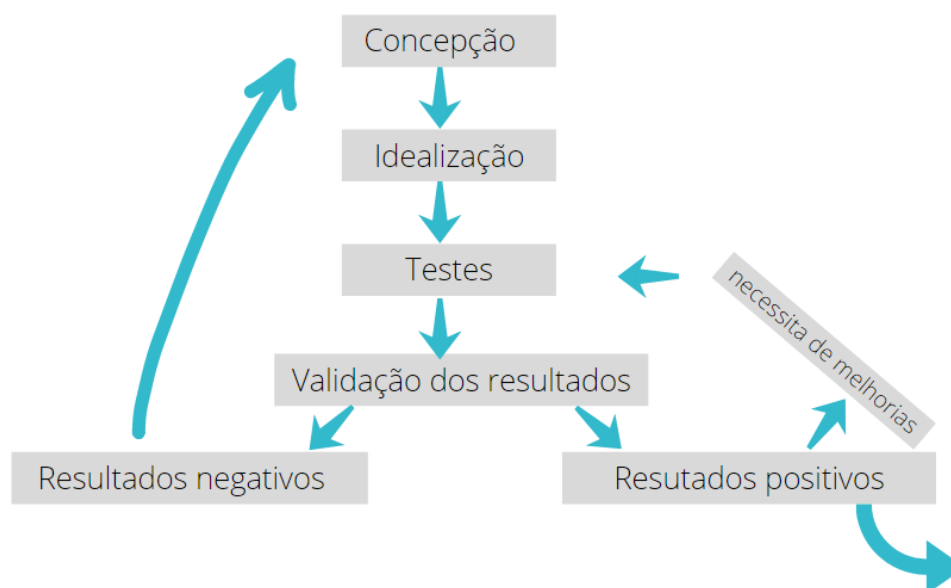
Nesta seção está descrita a base científica para definir o processo de desenvolvimento do jogo, com foco na metodologia utilizada desde a concepção até o jogo estar disponível para testes para os jogadores.

3.1.1 Processo de design centrado no jogador (*Playcentric Design Process*)

O *Playcentric Design Process* é uma abordagem para desenvolvimento de jogos que foca na experiência do jogador, a fim de envolvê-lo no design do processo com iterações significativas, personagens e histórias mais complexas (Fullerton, 2006). A proposta do estudo de Fullerton (2014) corrobora com essa abordagem e consiste em focar na experiência do jogador pelo método de iteração, na qual cada etapa é feita, testada e revisada durante todo processo do jogo até a sua concepção final.

A dinâmica de iteração concebe, idealiza, testa e valida a ideia (FULLERTON, 2014). Este processo é ilustrado na Figura 9.

Figura 9 – Processo de iteração



Fonte: Adaptado de Fullerton (2014)

As etapas descritas na Figura 9 definem o processo de iteração, seguindo a seguinte dinâmica: é iniciada a formalização e concepção das ideias, idealização, teste da ideia com base na experiência do jogador e, conseqüentemente, avaliar e priorizar os resultados obtidos através dos testes.

Caso os resultados sejam negativos, é necessário retornar para o processo de concepção. Caso sejam positivos, mas necessitem de melhorias, deve-se estabelecer as melhorias, iniciar novamente a fase de testes e, para finalizar, fazer a avaliação e priorização dos resultados. Já nos casos dos resultados que não possuam a necessidade de alterações, o processo de iteração estará concluído.

O objetivo do *Playcentric Design Process* é envolver o jogador em todo processo de desenvolvimento do jogo, desde a concepção até a conclusão. Segundo Zimmerman (2003), o processo de iteração é cíclico e composto por prototipagem, teste, análise e refinamento do trabalho em andamento. As decisões de *design* são baseadas na experiência do protótipo, que apesar de não simplificarem os recursos de desenvolvimento, tornam o produto final mais robusto e bem sucedido (ZIMMERMAN, 2003).

Como pode ser visto na Figura 4, o *Playcentric Design Process* possui quatro estágios que se utilizam do método de iteração. Para etapa de **Concepção**, inicia-se o processo de elaboração das ideias. Nesta etapa, utilizam-se ferramentas para auxílio no processo criativo, tais como: *brainstorm*, mapa mental, criação de listas, cartões ideias, entre outros. Além disso, são definidas as características do jogo, formalizadas as ideias e analisada a viabilidade do jogo (FERNANDES; LUCENA; ARANHA, 2018).

Na próxima etapa, inicia-se a **Pré-produção**. Nesta etapa, verifica-se a viabilidade do jogo por meio de prototipações físicas, digitais e documentos GDD e TDD. A prototipação física pode ser realizada com papel, desde que o foco seja o jogador. É possível testar, entender e validar conceitos do jogo. Em seguida, para a prototipação digital, as características gerais do jogo já estarão validadas e possuirão maturidade para transmitir as ideias para um jogo digital, tornando-o mais atraente para testar e receber um *feedback*. A produção dos documentos GDD e TDD fornecem informações gerais da jogabilidade, criação do jogo, modo de implementação e também documenta as etapas de concepção após estarem consolidadas (FULLERTON, 2014).

Na etapa seguinte, inicia-se a **Produção** que, com todas as etapas de planejamento estabelecidas, inicia-se o desenvolvimento do jogo. Os artefatos desenvolvidos nas outras etapas são revisados e refinados (FULLERTON, 2014).

Por fim, na última etapa, inicia-se a **Garantia da Qualidade**. Nesta etapa o jogo é testado e disponibilizado ao público-alvo (CHANDLER, 2009). A parte de interação é contínua enquanto o jogo estiver em mercado. Os testadores podem ser divididos em públicos-alvo para assegurar *feedbacks* e possíveis melhorias e continuidades do jogo (FULLERTON, 2014).

Na próxima seção está descrito o processo de desenvolvimento de *software* e as ferramentas utilizadas para seu embasamento.

3.2 Processo de desenvolvimento de software

Nesta seção está descrita a pesquisa utilizada para definir o processo de desenvolvimento do jogo, com foco nas metodologias de desenvolvimento ágil.

3.2.1 Scrum

Segundo Schwaber e Beedle (2002), o *Scrum* foi fundamentado nas teorias empíricas de processo, na qual a experiência e tomadas de decisões formam o conhecimento. *Scrum* é uma metodologia ágil que se apoia em três pilares: transparência, inspeção e adaptação (SCHWABER; BEEDLE, 2002). Além disso, é baseada em princípios da *extreme Programming* (XP), equipes pequenas, requisitos estáveis ou desconhecidos e iterações curtas (SOARES, 2004). Os eventos presentes no *Scrum* são encontros formais e obedecem a algumas definições, segundo Soares (2004).

A *Sprint* é o espaço de tempo de um mês ou menos, para produção de artefatos e desenvolvimento. Durante uma *Sprint* não ocorrem mudanças que alterem o seu plano, porém podem haver negociações caso existam alterações na produtividade. As *Sprints* podem ser canceladas caso o objetivo se torne obsoleto (SOARES, 2004). O planejamento

da *sprint* inicia-se a *Sprint*, que resulta em um trabalho coletivo com os envolvidos e define os próximos passos (SOARES, 2004).

Na reunião diária o objetivo é acompanhar o progresso da *Sprint*, a fim de ajustar e auxiliar o trabalho planejado. A reunião diária é um evento de quinze minutos com o time *Scrum* e desenvolvedores, que ocorre no mesmo horário e em dias úteis (SOARES, 2004). Na revisão da *Sprint*, inspecionam-se os resultados alcançados da *Sprint*, e caso necessário, são apontadas as melhorias necessárias (SOARES, 2004). Na retrospectiva da *Sprint* são planejadas formas de aumentar a produtividade e eficiência do time *Scrum*, por meio de análises do que foi desenvolvido até o momento (SOARES, 2004).

O time *Scrum* fundamentalmente é uma equipe de poucas pessoas que não possuem subequipes ou hierarquias (SOARES, 2004). Existem três cargos específicos dentro do *Scrum*: desenvolvedores, *product owner* e *scrum master*.

Os desenvolvedores possuem quatro competências dentro de uma time *Scrum*, segundo Soares (2004). Essas competências são: criar artefatos referentes a planejamento da *Sprint*, garantir qualidade, adaptar o planejamento da *Sprint* e responsabilizar o trabalho junto aos outros desenvolvedores.

O *product owner* é o responsável pela gestão e organização do time *Scrum*, reforçando a melhoria de comunicação, priorização e refinamento de itens do *backlog* do produto (SOARES, 2004).

O *scrum master* é o responsável pela implementação do *Scrum*, seguindo o “Guia do *Scrum*”. As competências do *Scrum Master*, segundo Soares (2004), são: treinar membros para multifuncionalidades, mapear impedimentos no Time *Scrum* e assegurar a produtividade nos eventos marcados.

A documentação ou artefatos *Scrum* desenvolvidos pela equipe representam o trabalho e o valor criado (SOARES, 2004). Os artefatos possuem objetivos definidos e possibilitam melhorias, sendo três existentes: *backlog*, *backlog da Sprint* e Incremento (SOARES, 2004).

O *backlog* do produto refere-se a um catálogo que ordena o que deve ser feito durante o processo de criação do *software*. Suas características são: funções, requisitos e podem ser realizadas futuras alterações (SOARES, 2004). O *backlog da sprint* é o conjunto de itens descritos no *backlog* do produto que são produzidos na *Sprint* (SOARES, 2004). O incremento é o documento que contém todos os itens do *backlog* do produto e devem ser completados durante a *Sprint*. Desse modo, ao final de cada *Sprint* um novo artefato de incremento deve estar pronto (SOARES, 2004).

3.2.2 Kanban

Kanban é um *framework* de gestão de mudanças para o desenvolvimento de *software* de métodos ágeis, que se utiliza de visão menos prescritiva que o *Agile* e tem se tornado uma extensão de métodos XP e *Scrum* (BOEG, 2010). O *Kanban* é derivado do método *Lean* e apresenta um quadro de atividades, a fim de visualizar o fluxo de trabalho de uma determinada organização (ALEXANDRE; JUNIOR, 2020). Segundo Alexandre e Junior (2020), o método para se utilizar o *Kanban* segue conceitos do *Statik*, uma abordagem de pensamentos sistemáticos que consiste em cinco passos:

1. Entender e descrever o que motiva a mudança;
2. Analisar demanda e capacidade de serviço;
3. Mapeamento de fluxo de trabalho;
4. Desenho do quadro do *Kanban*;
5. Reunião de início de projeto.

Estas etapas sugerem informações de como seguir o modelo *Kanban* para concretizar o quadro e ter controle de tarefas a serem desenvolvidas (BOEG, 2010). Ainda, as métricas que apoiam a estrutura do *Kanban* permitem um acompanhamento do projeto, análise de tomada de decisões e auxiliam as próximas tarefas (BOEG, 2010).

O processo metodológico do *Kanban*, no contexto de *software*, é um avanço crescente de mudanças que limitam o trabalho e criam um sistema de tração, em que o novo trabalho só pode ser iniciado quando o anterior estiver concluído (BORTOLUCI et al., 2017).

A confecção do quadro descreve a aplicação do método *Kanban*. A definição de estrutura do quadro aplica-se a classe de serviço a ser desempenhado (BOEG, 2010). Na Figura 10 é apresentado um modelo genérico de colunas para o quadro.

Figura 10 – Painel genérico do Kanban

Backlog do Produto	Backlog da Sprint	Próximo	Análise e Design		Desenvolvimento		Teste		Documentação		Aceitação	Implementação	Feito
			Fazendo	Feito	Fazendo	Feito	Fazendo	Feito	Fazendo	Feito			
█ █ █ █ █	█ █ █	█ █	█ █	█	█ █ █	█	█ █	█ █	█ █ █	█ █	█ █	█	

Fonte: Adaptado de [Mahnic \(2014\)](#)

Segundo [Mahnic \(2014\)](#), o quadro representado na Figura 10 abrange uma gama completa de atividades, sendo elas: *backlog*, *backlog da sprint*, próximo, análise e *design*, desenvolvimento, teste, documentação, aceitação, implementação e definição de pronto.

O *backlog* do produto contém todas as histórias dos usuários que se tem conhecimento. O *backlog da Sprint* contém as histórias do usuário que se desenvolvem na *Sprint*. O “próximo” contém um limitado número de histórias de usuário, que possuem uma maior prioridade. A análise e *Design* divide-se em duas colunas, “em desenvolvimento” e “concluído”, refere-se a diluição de uma história de usuário em micro tarefas. O desenvolvimento refere-se a tarefa ou história de usuário, que no espaço de tempo está sendo desenvolvida. O desenvolvimento é considerado concluído quando todas as unidades são testadas.

No teste são evidenciadas as tarefas que não estão em desenvolvimento, mas que passam por testes para alcançar o trabalho feito. A documentação contém novas funcionalidades da história de usuário que requerem a documentação. A aceitação corresponde ao teste final de aceitação realizado pelo proprietário do produto. A implementação contém histórias de usuário que estão sendo implantadas. O “pronto” contém histórias aceitas e disponibilizadas.

Na próxima seção estão descritas algumas definições de IHC, consideradas oportunas para este trabalho.

3.3 Interação humano-computador

Nesta seção serão apresentados os conceitos básicos que envolvem IHC, pertinentes ao trabalho. São eles: interação, interface, usabilidade, *User eXperience* (UX) ou, eXperiência do Usuário, acessibilidade e comunicabilidade, sendo o tópico de usabilidade o

principal para a presente pesquisa.

Segundo [Barbosa et al. \(2021\)](#), interação no contexto de IHC é um processo de manipulação, comunicação, conversa, troca e influência entre o usuário e o sistema. A interação usuário-sistema possui quatro perspectivas: sistema, parceiro de discurso, ferramenta e mídia ([KAMMERSGAARD, 1988](#)).

Para [Moran \(1981\)](#), interface de um sistema interativo é *“toda a porção do sistema com a qual o usuário mantém contato físico ou conceitual durante a interação”*. O contato físico refere-se a parte de *hardware*, como os dispositivos de entrada e os de saída, e o contato conceitual envolve a percepção do usuário sobre as ações realizadas pelos dispositivos de entrada e saída, no sistema. A interface do sistema com o usuário determina o que ele pode fazer, como e a ordem de ação.

A usabilidade possui várias definições, sendo [ISO-9241-210:2019 \(2019\)](#) a mais utilizada. É conceituada como *“o grau em que um produto é usado por usuários específicos para atingir objetivos específicos com eficácia, eficiência e satisfação em um contexto de uso específico”*. [Nielsen \(1994b\)](#) define cinco critérios de usabilidade que todo sistema deve ter: facilidade de aprendizado, facilidade de recordação, eficiência, segurança no uso e satisfação do usuário.

Segundo a [ISO-9241-210:2019 \(2019\)](#), UX consiste em *“percepções e respostas de uma pessoa que resultam do uso ou da antecipação do uso de um produto, sistema ou serviço”*. Ainda que existem outras, esta definição da *International Organization for Standardization (ISO)* é a mais utilizada na literatura.

[Forlizzi e Battarbee \(2004\)](#) desagregam a UX em três grupos: produto, usuário e interação. Além disso, identificaram três tipos de interação: fluentes, cognitivas e expressivas, que podem gerar três tipos de experiência, sendo: experiência, uma experiência e co-experiência. Outros autores, [Hassenzahl e Tractinsky \(2006\)](#) identificaram três perspectivas, que combinadas formam a UX: experiencial; emoção e afeto; e instrumental. Em estudo posterior, [Hassenzahl \(2010\)](#) descreveu cinco principais propriedades da experiência do usuário: subjetiva, holística, situada, dinâmica e positiva.

Segundo [Barbosa et al. \(2021\)](#), acessibilidade é *“a capacidade de o usuário acessar o sistema para interagir com ele, sem que a interface imponha obstáculos”*. A acessibilidade busca a inclusão dos usuários que utilizam o sistema, ou seja, não considera as limitações e trata de forma igual todos os usuários. Já a comunicabilidade, é definida por [Souza \(2005\)](#) como *“a 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”*. O *design* deve conseguir comunicar ao usuário para qual tipo de público o sistema se destina, para que serve, a vantagem de utilizá-lo e como funciona.

Na próxima seção estão descritas as heurísticas de usabilidade de Nielsen e como

se relacionam ao trabalho.

3.4 Heurísticas de Usabilidade de Nielsen

Nesta seção estão descritos os tópicos que envolvem as heurísticas de usabilidade de Nielsen, que contemplam este trabalho. Na primeira Subseção 3.4.1 está descrito o contexto de aplicação das heurísticas no trabalho, na Subseção 3.4.2 está descrito a definição das heurísticas de Nielsen de interface e na Subseção 3.4.3 está descrito a avaliação da heurística de interface.

3.4.1 Contexto de aplicação

A avaliação das Heurísticas segundo Nielsen (1994a), é um método de engenharia de usabilidade para examinar interfaces e identificar problemas. O conjunto original de heurísticas de usabilidade utilizado em vários estudos iniciais foi desenvolvido com o objetivo principal de tornar o método fácil de ensinar e adaptável para outras aplicações (NIELSEN, 1994a).

3.4.2 Definição das heurísticas de Nielsen de interface

As heurísticas de Nielsen são definidas por dez princípios gerais para o *design* de interface do usuário e estão descritas a seguir:

1. **Visibilidade do status do sistema (*Feedback*):** O sistema deve sempre manter os usuários informados, por meio de *feedbacks* satisfatórios em tempo hábil (NIELSEN, 2005).
2. **Combinação entre o sistema e o mundo real (*Metáfora*):** O sistema deve utilizar a terminologia apropriada ao usuário (BORGES; RAPKIEWICZ; FEIJÓ, 2012).
3. **Controle e liberdade do usuário (*Navegação*):** O sistema deve permitir um suporte para desfazer e refazer ações quando o usuário emprega uma função errada (NIELSEN, 2005).
4. **Consistência e padrões (*Consistência*):** O sistema deve seguir convenções e padrões da plataforma para que sejam minimizadas as dúvidas do usuário (NIELSEN, 2005).
5. **Prevenção de erros (*Prevenção*):** O sistema deve ter um *design* que evite erros do usuário, como mensagens de confirmação antes de ações que comprometem, por exemplo (NIELSEN, 2005).

6. **Reconhecimento em vez de recordação (Memória):** O sistema deve minimizar a recordação do usuário e facilitar o reconhecimento de objetos (NIELSEN, 2005).
7. **Flexibilidade e eficiência de uso (Eficiência):** O sistema deve trazer informações como abreviações e teclas de funções por meio de atalhos para facilidade do usuário na navegação (BORGES; RAPKIEWICZ; FEIJÓ, 2012).
8. **Design estético e minimalista (Design):** O sistema deve conter diálogos, informações relevantes e necessárias (NIELSEN, 2005).
9. **Ajude os usuários a reconhecer, diagnosticar e se recuperar de erros (Recuperação):** As mensagens de erro devem ser expressas em linguagem compreensível, indicar a ação e recomendar uma resposta construtiva (NIELSEN, 2005).
10. **Ajuda e documentação (Ajuda):** A sistema deve ser intuitivo a fim de evitar a necessidade de ajuda, porem é indispensável a documentação e ajuda do sistema de forma clara ao usuário (BORGES; RAPKIEWICZ; FEIJÓ, 2012).

3.4.3 Avaliação heurística de interface

A avaliação heurística de interface é um método de eficiência para apontar erros do sistema, por meio da experiência dos avaliadores que buscam encontrar potenciais problemas de interface no sistema utilizado (MACIEL et al., 2004). Os princípios para avaliação do sistema são cinco: quando deve ser utilizada, avaliadores, eficiência, preparação e método (NIELSEN, 1994b).

O primeiro princípio, de avaliação do sistema, representa quando pode ser empregada e corresponde a todas as fases do ciclo de desenvolvimento do *software*, mas aconselha-se utilizar nas fases iniciais (MACIEL et al., 2004). O segundo princípio refere-se aos avaliadores e são recomendados ao menos dois, com competência em usabilidade, evitando projetistas da interface do produto (MACIEL et al., 2004). O próximo princípio é o de eficiência e propõe três avaliações com cinco avaliadores, para melhor distribuição de custos (MACIEL et al., 2004).

O princípio de preparação permite ao avaliador conduzir o teste sem ajuda adicional, mas ressalta que é necessário o suporte, caso seja uma aplicação específica para o uso do sistema (MACIEL et al., 2004). O último princípio refere-se ao método, e orienta as fases que devem ser seguidas pelos avaliadores e pela equipe de testes. São elas: instrução específica para o uso do sistema, orientação para execução dos testes, consolidação dos problemas apontados e análise dos problemas por severidade, documentação e resultados (MACIEL et al., 2004).

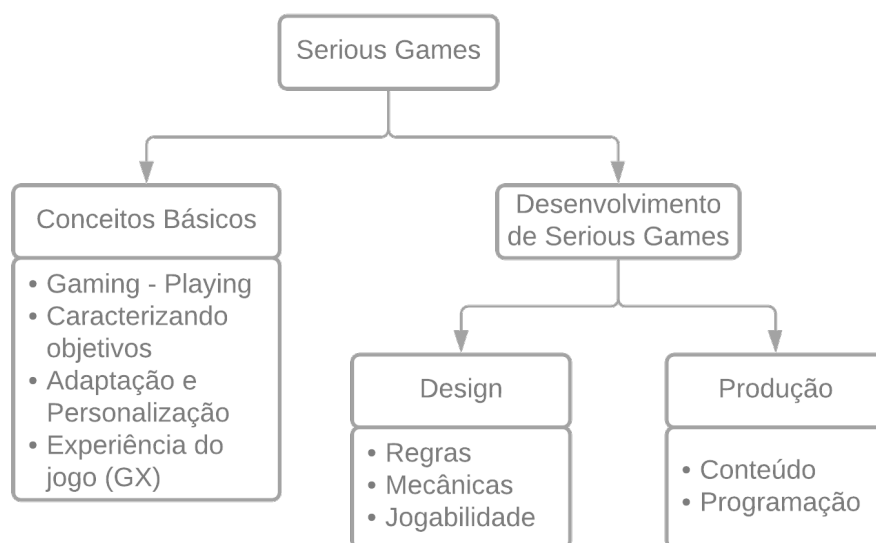
Na próxima seção serão apresentados os conceitos de jogos para ensino e suas características.

3.5 Jogos para ensino

Nesta seção é apresentado a definição de jogos voltados para ensino, que fazem parte de um subgrupo dos jogos sérios (*serious games*), e também sobre o processo de ensino-aprendizagem de jogos.

Segundo Fullerton (2014), jogos são “um sistema fechado e formal que engaja jogadores em conflitos estruturados e resolve suas incertezas em um resultado desigual”. Os jogos possuem limites com suas próprias regras, contextos e elementos. Dentro dos subgrupos existentes nos jogos, tem-se o *serious games* ou jogos sérios, como um tipo de jogo. De acordo com Dörner et al. (2016), *serious game* é “um jogo digital criado com a intenção de entreter e de alcançar ao menos um objetivo adicional”. Além disso, são caracterizados não mais pelos desenvolvedores, e sim pelos jogadores do jogo, mas sempre voltados para à aprendizagem (DÖRNER et al., 2016). Na Figura 11, pode-se visualizar os termos básicos de elementos presentes nos *serious games*.

Figura 11 – Terminologias dos *Serious Games* (visão geral dos termos básicos)



Fonte: Adaptado de Dörner et al. (2016)

A Figura 11 apresenta os conceitos básicos de um *serious game* e os elementos que estão presentes dentro do desenvolvimento de jogos. Dentro do desenvolvimento, tem-se o *design* e produção do jogo. No *design* são elaboradas as regras, mecânicas e a jogabilidade, e na produção do jogo, o conteúdo é criado juntamente com a programação.

Existem especificidades dentro dos *serious games*, entre elas, os jogos educacionais, que constituem o foco do trabalho. Dörner et al. (2016) conceitua jogos educacionais como “um subgrupo dos *serious games*, abordando o setor da educação formal de escolas

primárias até as universidades, treinamentos vocacionais, e treinamento para local de trabalho colaborativo”. É importante destacar a diferença entre jogos de aprendizagem e jogos educacionais; os de aprendizagem tem foco na informalidade e os jogos educacionais são direcionados para a aprendizagem formal em instituições de ensino e são uma das áreas de IHC (DÖRNER et al., 2016).

Na próxima seção são apresentadas pesquisas sobre trabalhos similares ao proposto neste documento.

3.6 Trabalhos correlatos

Nesta seção são apresentados quatro trabalhos similares sobre jogos digitais que ensinam sobre as heurísticas de usabilidade de Nielsen. As pesquisas levantadas são referentes aos jogos *Usability*, *UsabilityGame*, *MACteaching*, *PersonaDesignGame*.

O *Usability* é um jogo *singleplayer* que faz analogias à uma cidade com dez problemas, ligados as dez heurísticas de usabilidade de Nielsen. O jogador tem o papel de inspetor e deve ajudar a identificar esses problemas, escolhendo uma das heurísticas para resolver cada um deles (FERREIRA et al., 2014).

O *UsabilityGame* também é um jogo *singleplayer*, no qual o jogador possui o papel de engenheiro de usabilidade, simulando a situação de uma empresa fictícia de desenvolvimento de *software*, e deve resolver problemas relacionados a *design* de interação (SOMMARIVA et al., 2011).

O *MACteaching*, assim como os dois anteriores, é um jogo *singleplayer* que propõe a aprendizagem por meio de analogias. O objetivo do jogador é ajudar a salvar uma cidade de uma bruxa e resolver o máximo de problemas, relacionando com treze respostas do Método de Avaliação de Comunicabilidade (MAC) (BRITO et al., 2016).

O *PersonaDesignGame* é um jogo *singleplayer* de perguntas e respostas de gênero educacional sobre personas. Neste jogo, o jogador pode aprender o conteúdo sobre a técnica de personas, exercitá-lo, e, por fim, ter um exemplo prático de como se construir uma persona (JÚNIOR, 2021).

Com base nestes jogos pesquisados, foi possível identificar pontos positivos destes jogos, como analogias as heurísticas de usabilidade de Nielsen, resolução de problemas e interações de usuários. Também foi possível perceber que estes jogos necessitam de melhorias de usabilidade e de jogabilidade a fim de dar uma experiência mais imersiva ao jogador. Estas melhorias listadas podem ser vistas na Tabela 1. Deste modo, considera-se que os aspectos de IHC podem ser melhor trabalhados entre o relacionamento jogo e jogador.

Na próxima seção estão descritas as tecnologias escolhidas para este trabalho, que

apoiam o documento de TDD apresentado na Seção 4.3.

3.7 Tecnologia

Nesta seção estão descritas as tecnologias escolhidas para o desenvolvimento do projeto, sendo: Docker¹, PostgreSQL², Node.js³, Express.js⁴, Vue.js⁵, Heroku⁶ e Vercel⁷. As tecnologias foram escolhidas com base em conhecimentos prévios e afinidades dos autores desta pesquisa. As comparações das vantagens e desvantagens, juntamente com a justificativa das escolhas das tecnologias, podem ser encontradas no Apêndice A.

Docker é uma ferramenta para encapsulamento de artefatos, distribuídos para aplicações em escala de ambiente de desenvolvimento (VITALINO; CASTRO, 2016). A ferramenta utiliza modelo de contêiner que permite que um conjunto de tecnologias integrem o espaço de virtualização, além de fornecer infraestrutura de gestão para as aplicações, evitando erros de ambientação entre as partes desenvolvedoras (FERREIRA, 2021).

O PostgreSQL é um sistema de gerenciamento de banco de dados relacional em código aberto e orientado a objetos. Os atributos do PostgreSQL são comuns a banco de dados de grande porte, possuindo versatilidade, segurança e gratuidade; por isso é uma ferramenta muito utilizada na comunidade acadêmica e também nas empresas que buscam um baixo custo (MAGALHÃES et al., 2015).

Node.js é um *framework* para programações orientadas a objetos que executa servidores implementados em JavaScript. Esta ferramenta permite a comunicação por meio de eventos entre servidor/cliente de uma maneira que difere da convencional. Sua aplicação permite ao cliente não aguardar a resposta de uma chamada para iniciar um novo evento (EVANGELISTA, 2013). Já o Express.js é um *framework web* baseado nos *frameworks* Node.js e Connect. Os componentes do Express.js são denominados de *middleware* e formam sua base de construção. Além disso, os sistemas Express.js possuem facilidade em configurações e permite aos desenvolvedores a escolha de bibliotecas para cada projeto (MARDAN, 2018).

De acordo com Macrae (2018), Vue é “*Framework JavaScript atual que está amadurecendo junto à comunidade open source e diante disso aumentando sua popularidade*”. Sua concepção permite o desenvolvimento de aplicativos complexos baseados na *web* que alteram seus dados sem atualizações de outras páginas subsequentes. Sua arquitetura constitui a *View* de um modelo *Model-View-Controller* (MVC) e abstrai o *Document Ob-*

¹ <<https://www.docker.com/>>

² <<https://www.postgresql.org>>

³ <<https://nodejs.org/en/>>

⁴ <<https://expressjs.com/pt-br/>>

⁵ <<https://vuejs.org>>

⁶ <<https://https://www.heroku.com>>

⁷ <<https://https://vercel.com/>>

ject Model (DOM), que visualiza os elementos em uma estrutura de árvore (MACRAE, 2018).

De acordo com Neto (2020), Heroku é uma plataforma de serviços com foco na aplicação de componentes para implantação de uma aplicação de *software*; sua estrutura permite cadastrar variáveis de ambientes, conectar a banco de dados e hospedá-los em um servidor gratuito.

Vercel é uma ferramenta de implantação automática que possui uma estrutura capaz de auxiliar que aplicações sejam conectadas em um único repositório para maior escalabilidade (AGUIAR et al., 2020).

Outras tecnologias que foram utilizadas no apoio da execução deste trabalho estão relacionadas a comunicação e a organização de código e de arquivos. No auxílio da comunicação foram selecionadas as ferramentas: Discord⁸, Microsoft Teams⁹ e Telegram¹⁰. No auxílio da organização do código e dos arquivos as ferramentas escolhidas foram: GitHub¹¹ e Google Drive¹², respectivamente. Para as tecnologias relacionadas a comunicação e a organização de código e de arquivos, as escolhas foram feitas com base em utilizações prévias dos pesquisadores deste trabalho. No próximo capítulo, encontra-se a concepção do jogo e os resultados alcançados neste trabalho.

⁸ <<https://discord.com/>>

⁹ <<https://www.microsoft.com/pt-br/microsoft-teams/free>>

¹⁰ <<https://telegram.org/>>

¹¹ <<https://github.com/>>

¹² <<https://www.google.com/intl/pt-BR/drive/>>

4 Concepção do jogo

Neste capítulo são apresentados os resultados alcançados com base nos objetivos e metodologia de desenvolvimento de *software* e a metodologia de desenvolvimento de jogo. Nas seções seguintes são apresentados os resultados das etapas do ciclo de vida do jogo e do GDD e TDD.

4.1 Ciclo de vida do jogo

O desenvolvimento do jogo iniciou-se com a revisão bibliográfica e definição do escopo, apresentados no Capítulo 2. O ciclo de vida do jogo é apoiado no processo do *Playcentric Design Process* e diretrizes de Engenharia de *Software*, que foram detalhados no Capítulo 3. Foi possível elicitar requisitos por meio da revisão bibliográfica na área de pesquisa de jogos para ensino-aprendizagem e também no processo *Playcentric Design Process* com a criação de *brainstorm* e protótipos de baixa e alta fidelidade.

Para o desenvolvimento do jogo utilizou-se o processo iterativo. Os protótipos desenvolvidos foram refinados com base nas avaliações de Persona, coletadas do trabalho do autor Silva (2021). O método de avaliação utilizado foi o modelo do MEEGA+ que pode ser visto no Anexo A.

A partir da evolução dos protótipos foram elaborados elementos e características, também foram definidos requisitos funcionais e não funcionais, jogabilidade, arquitetura e tecnologias utilizadas no desenvolvimento do jogo. Sendo assim, após a etapa de desenvolvimento, foi realizado a implantação para que os jogadores possam acessar o jogo. Estes artefatos estão presentes no GDD e TDD, nas Seções 4.2 e 4.3 respectivamente.

4.2 Documento de projeto do jogo (*Game design document*)

Nesta seção é apresentado o GDD, documento incremental, conforme a evolução do trabalho, que dispõe sobre as características e elementos gerais do jogo, jogabilidade, protótipo de baixa fidelidade e protótipo de alta fidelidade.

4.2.1 Características gerais

O jogo desenvolvido neste trabalho teve como foco as heurísticas de usabilidade de Nielsen e é classificado como um jogo do gênero educacional. O jogo é do tipo *perguntas e respostas*, *singleplayer*, em primeira pessoa e foi 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 alcança o próximo nível.

4.2.2 Jogabilidade (*Gameplay*)

O objetivo é fazer o jogador avançar níveis, por meio de respostas corretas de múltiplas escolhas e verdadeiro ou falso. O fluxo utilizado para execução do jogo seguiu passos pré-determinados:

1. Ao acessar o menu principal, o jogador opta por iniciar o jogo, visualizar a tabela de *ranking* ou acessar a tela “sobre nós”;
2. Ao iniciar o jogo é apresentada uma breve contextualização do ambiente;
3. Em seguida, são exibidos os projetos que o personagem avaliará;
4. Ao selecionar e ser apresentado ao projeto, o jogador responde uma pergunta aleatória, que pode ser dos tipos: “verdadeiro ou falso” ou “múltipla escolha”;
5. É indicado se o jogador respondeu corretamente a pergunta e a pontuação é atualizada;
6. Caso a resposta esteja errada, apresentará a resposta certa e o porquê;
7. Após responder as vinte questões possíveis, caso o jogador atinja a pontuação necessária, é apresentado o próximo nível;
8. Caso o jogador não atinja a pontuação necessária, poderá reiniciar o nível;
9. Caso o jogador acesse o menu do ranking, será informada sua pontuação e uma lista com os maiores pontuadores e percentuais de acertos.

O jogo foi delineado em três fases, associadas ao desenvolvimento de cada jogador dentro do jogo. A primeira fase possui perguntas de conhecimento básico, na segunda fase as perguntas são de nível mediano, e por fim, a fase com perguntas de nível de conhecimento avançado. Para escopo de desenvolvimento, apenas a primeira fase com três níveis de dificuldade foi criada; sendo o primeiro nível com perguntas fáceis, o segundo com perguntas de dificuldade mediana e o terceiro com perguntas difíceis - dentro do conhecimento mais simples sobre as heurísticas de usabilidade de Nielsen.

A dificuldade do jogo é apresentada a partir de cada pergunta, em diferentes níveis e fases. O jogador deve atingir 30 (trinta) pontos para passar à fase do primeiro nível e 100 (cem) pontos para passar do segundo nível. A pontuação das questões difere de acordo com a dificuldade que está associada ao nível escolhido. Existe um banco de questões para

cada nível e fase, a fim de evitar que o jogador repita a mesma combinação de questões e a mesma ordem de perguntas, este ponto soluciona o problema das fases dinâmicas apresentadas na Tabela 1.

4.2.3 Elementos do jogo

Nesta subseção são abordados os elementos que caracterizam o jogo, a saber: personagem, telas, perguntas e *ranking*. A evolução do aprendizado do jogador foi o principal foco no desenvolvimento do jogo.

4.2.3.1 Personagem

O personagem principal escolhido é um estudante da Universidade de Brasília que ajuda outros alunos com seus conhecimentos em Heurísticas no âmbito de IHC, por meio de projetos desenvolvidos dentro da universidade. O sexo do personagem será decisão do jogador, que pode escolher uma nome de até 3 letras, afim de aproximar o jogador aos jogos *arcade* de fliperama, e imagem que se refere ao sexo, como mostrado na Figura 12, retirada do jogo real. A escolha de personagem foi implementado e está presente na Tabela 1 como uma das melhorias que foram implementadas.

Figura 12 – Imagem de seleção de personagem



Fonte: Elaboração própria (2022)

Estas decisões apresentadas nas Figura 12 foram inspiradas por jogos do estilo *arcade*, em que o ambiente destes jogos é de gráficos 8 bits, com um ranking simples e reduzido que remetem a uma competitividade e curiosidade.

4.2.3.2 Perguntas

As perguntas, *feedbacks* e a classificação do nível das perguntas são de elaboração própria e iniciou-se através dos estudos sobre os dez princípios das heurísticas de Nielsen

(Seção 3.4.2). Ao todo, o jogo possui 60 perguntas no banco de dados, divididas igualmente entre tipos, verdadeiro ou falso e múltipla escolha (30 perguntas de verdadeiro ou falso e 30 perguntas de múltipla escolha). Foram subdivididas entre os níveis fácil, médio e difícil. Cada nível possui 20 perguntas, no formato de cartões, fragmentadas entre tipo verdadeiro ou falso e tipo múltipla escolha (dez de cada). O banco de dados com todas as perguntas criadas para o jogo se encontra no Apêndice D.

Para as perguntas de tipo verdadeiro ou falso, é apresentada a definição de uma heurística e perguntado se a sentença está correta. Criou-se também casos fictícios de empresas que em seus sites possuem elementos que contradizem os princípios das usabilidades de Nielsen e perguntado se determinado princípio desta heurística é violada. Para as perguntas do tipo múltipla escolha, foram feitos recortes de sites e aplicativos que violavam os princípios das heurísticas de usabilidade proposta por Nielsen e disposto quatro opções ao jogador destas heurísticas.

Em cada nível, o jogo apresenta somente vinte perguntas do banco de dados, de modo randomizado, sendo dez tipo verdadeiro ou falso e dez tipo múltipla escolha. Com esta dinâmica de perguntas sendo apresentadas de forma randômica e com muitas perguntas por nível, a melhoria de fases dinâmicas apontada na Tabela 1 foi implementada. Os níveis que o jogador não alcançar a pontuação necessária ficam visivelmente bloqueados. A pontuação e a quantidade de perguntas acertadas e tentadas também estão disponíveis para o jogador visualizar durante o jogo. A cada resposta certa, a pontuação é apresentada na tela e acrescentada à pontuação da sessão automaticamente. Para respostas erradas, é apresentada a resposta correta e uma explicação do porquê.


As perguntas têm o nível de dificuldade de acordo com a evolução do jogador. Os elementos levantados para a confecção das perguntas foram: os objetivos e metas estabelecidas para o nível do projeto, dificuldade da pergunta e tipo de pergunta, verdadeiro ou falso e múltipla escolha.

Para classificação das perguntas quanto à dificuldade, foi utilizada a metodologia do *Planning Poker*, que consiste nos integrantes da equipe envolvida no trabalho estimarem por meio de pontuação/classificação as *issues* do projeto (GRENNING, 2002). Neste trabalho, utilizou-se a metodologia para que os autores classificassem as perguntas quanto ao nível de dificuldade de cada uma delas: fácil, médio ou difícil. Desse modo, os autores alcançaram um acordo, classificando as seis perguntas de verdadeiro ou falso e de múltipla escolha consideradas mais fáceis, mais difíceis e as de nível médio. A partir da classificação de seis perguntas, as outras 54 foram classificadas seguindo a metodologia, tendo como referência o que foi considerado fácil, médio ou difícil.

Nas Figuras 13 e 14, pode-ser observar exemplos de perguntas e dos *feedbacks* utilizados no jogo de múltipla escolha e verdadeiro ou falso, respectivamente.

Figura 13 – Exemplo de pergunta do tipo múltipla escolha

1) (FÁCIL) Qual heurística melhor representa a imagem abaixo:



a) **Visibilidade do status do sistema**

b) Compatibilidade entre o sistema e o mundo real

c) Controle e liberdade para o usuário

d) Consistência e padronização

FEEDBACK: A heurística que melhor representa a imagem é a de visibilidade do status do sistema. A imagem mostra as etapas do checkout em um processo na compra de um produto em uma loja virtual e essas etapas representam o status do processo de compra.

Fonte: Elaboração própria (2022)

A Figura 13 representa uma pergunta de múltipla escolha e nível de dificuldade fácil, junto ao *feedback* que é gerado em caso de resposta errada, por parte do jogador.

Figura 14 – Exemplo de pergunta do tipo verdadeiro ou falso

60) (DIFÍCIL) Quando você tenta acessar uma página de um site e ele te mostrar uma página com o erro 404, mostra que o site implementou a heurística de ajuda e documentação. Essa afirmação é verdadeira ou falsa? **F**

FEEDBACK: A heurística da afirmação é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros, ou seja, a afirmação estava errada. O erro 404 ajuda o usuário a saber que teve um erro e poder voltar à página anterior.

Fonte: Elaboração própria (2022)

A Figura 14 representa uma pergunta de verdadeiro ou falso e nível de dificuldade difícil, junto ao *feedback* que é gerado em caso de resposta errada, por parte do jogador.

Para a criação das perguntas foram utilizados os conhecimentos dos autores que foram adquiridos estudando sobre as heurísticas de Nielsen nas referências bibliográficas utilizadas neste trabalho, cinco textos em diferentes *sites* e em um *brainstorm*. Os textos

que foram utilizados como inspiração para criação de algumas perguntas são dos sites: UX Design (Gabriel Moma, 2017), Programmers (Gabrielle Pinhata, 2021), Alura (Marco Bruno, 2019), Vida de Produto (Henrique Carvalho, 2020), Vr Sys (VR SYS, 2019), Aela (Editorial Aela, 2019) e Vulpi (Wesley Simoes, 2018). A tabela contendo a relação das perguntas, o nível e suas referências podem ser vista no Apêndice D. O *brainstorm* realizado teve objetivo de complementar as perguntas, após terem sido criadas as perguntas com conceitos teóricos e as inspiradas nos sites.

4.2.3.3 Ranking

O *ranking* é um elemento importante do jogo Universidade das Heurísticas, pois é capaz de levar o espírito de competitividade dos jogadores para dentro do jogo. Os jogadores podem jogar até alcançar cem por cento de acurácia nos três níveis de dificuldade, para conseguirem alcançar o topo do *ranking*.

Explorar a competição é uma das características mais importantes que deve estar presente em jogo educativo digital (FALKEMBACH; GELLER; SILVEIRA, 2006). Dentro do jogo Universidade das Heurísticas é explorado esta característica através da classificação dos jogadores por porcentagem de questões acertadas e pontuação total, permitindo que o mesmo possa aprimorar seu desempenho em uma nova partida.

4.2.4 Protótipo de baixa fidelidade

Para elaborar o protótipo de baixa fidelidade, foi utilizada a técnica de prototipação em papel. O protótipo de papel pode ser visualizado em detalhes no Apêndice B ou na apresentação interativa dos desenhos, no site da Marvel¹. O modelo de como o jogo foi feito no protótipo de papel surgiu após um *brainstorm* com os autores. As telas do jogo foram desenhadas e um fluxo foi inserido na sequência. Posteriormente, o protótipo foi avaliado com quatro pessoas, sendo três personas identificadas pelo autor Silva (2021) e um estudante de engenharia de *software*. A decisão de utilizar personas originou-se da avaliação dos protótipos utilizadas pelo autor Júnior (2021) no jogo *PersonaDesignGame*. Na Subseção 4.2.4.1 está detalhado o processo de validação do protótipo de baixa fidelidade.

4.2.4.1 Processo de validação do protótipo de baixa fidelidade

Com o protótipo de papel desenvolvido, deu-se início ao processo de validação do mesmo, que teve como objetivo avaliar o protótipo desenvolvido a partir das metas de experiência do usuário, verificar problemas e apontar soluções para melhorias no protótipo de alta fidelidade. Para realizar a validação foi utilizado o processo proposto por Barbosa et al. (2021) para avaliação do protótipo de papel; o processo proposto possui cinco

¹ <<https://marvelapp.com/prototype/5953hgc>>

atividades: (1) preparação, (2) coleta de dados, (3) interpretação, (4) consolidação dos resultados e (5) relato dos resultados, sendo que cada uma delas possui uma ou mais tarefas a serem executadas.

A atividade de preparação possui quatro tarefas: (1) definir tarefa para os participantes executarem, (2) definir o perfil dos participantes e recrutá-los, (3) criar protótipo de papel da interface para executar as tarefas e (4) definir um teste piloto. A atividade de coleta de dados possui somente uma tarefa, sendo que cada usuário deve executar as tarefas propostas interagindo com os protótipos em papel, mediado pelo avaliador. A atividade de interpretação possui duas tarefas: (1) listar os problemas encontrados e (2) refinar os protótipos em papel para resolver os problemas mais simples. A atividade de consolidação dos resultados também possui duas tarefas: (1) priorizar a correção dos problemas não resolvidos e (2) sugerir correções. Por fim, a atividade de relato dos resultados possui uma tarefa que é a de relatar os problemas encontrados e sugestões de correção.

No processo de validação do protótipo de papel foi realizada somente uma iteração ao invés de duas iterações, como propõem [Barbosa et al. \(2021\)](#), devido ao pouco tempo que se tinha para executar o projeto. Foi realizado um novo processo de iteração com o protótipo de alta fidelidade, que pode ser visto na Seção [4.2.5.1](#).

Na atividade de preparação, a primeira tarefa feita consistiu na criação do protótipo, e em seguida, foram definidas as tarefas que os participantes executaram, ambas as tarefas podem ser vistas no Apêndice [B](#). Logo após, foram definidos e recrutados os participantes, que foram as três personas identificadas por [Silva \(2021\)](#), que podem ser vistas no Anexo [B](#), e um estudante de engenharia de *software*; posteriormente, foi realizado a avaliação dos participantes.

Na atividade de coleta de dados os participantes executaram o teste mediado pelo avaliador. Logo após a realização de cada teste, a atividade de interpretação foi feita com o avaliador tendo feito a listagem de todos os problemas apontados pelos participantes.

Na atividade de consolidação dos resultados todos os problemas apontados foram colocados em uma tabela e na coluna ao lado dos problemas foi feita uma sugestão de correção de cada um deles por parte dos desenvolvedores. A atividade de relato dos resultados foi o fechamento das informações e publicação das mesmas neste trabalho, como pode ser vista na Tabela [4](#).

Tabela 4 – Problemas e sugestões de correção da validação do protótipo de baixa fidelidade

Problema	Sugestão de correção
Não dá para ordenar informações no ranking	Adicionar um filtro crescente e decrescente nas colunas de nome, pontos e questões
Não é possível fazer a escolha de personagem	Adicionar, ao iniciar o jogo, um quadro de seleção de personagem
Não é possível fazer a escolha do nome	Adicionar, ao iniciar o jogo, um quadro com o campo para escolha do nome
Jogo de perguntas e respostas monótono	Adicionar no banco de dados mais perguntas do que o usuário responderá em um jogo e randomizar a ordem de aparecimento delas

Fonte: Elaboração própria (2021)

Na Tabela 5 são apresentadas as notas dadas pelos participantes referentes às metas de experiência do jogador. As notas podem variar de um até cinco, sendo um a pior nota e cinco a melhor. Os participantes P1, P2 e P3 são as personas primária, secundária e suplementar, respectivamente, que foram definidas no trabalho de [Silva \(2021\)](#), e o U1 é o estudante de engenharia de *software*.

Tabela 5 – Avaliação do protótipo de baixa fidelidade

Meta de Experiência	U1	P1	P2	P3	Pontuação Total	Porcentagem Total
Confiança	4	4	4	4	16	80%
Relevância	3	4	3	2	12	60%
Satisfação	3	4	2	3	12	60%
Atenção focada	2	3	4	2	11	55%
Desafio	3	2	3	3	11	55%
Diversão	3	3	2	2	10	50%

Fonte: Elaboração própria (2021)

Após a consolidação dos problemas apresentados na Tabela 4 e das notas apresentadas na Tabela 5, os problemas identificados e as metas de experiência com menor pontuação foram melhoradas no protótipo de alta fidelidade. O objetivo da avaliação do protótipo de baixa fidelidade de identificar problemas e apontar soluções no protótipo de papel foi atingido. O processo de avaliação do protótipo de alta fidelidade está apresentado na Subseção 4.2.5.1.

4.2.5 Protótipo de alta fidelidade

Para elaborar o protótipo de alta fidelidade, foi utilizada a ferramenta Figma, capaz de auxiliar essa construção com os elementos existentes na plataforma. O protótipo

pode ser visualizado no Apêndice C ou na apresentação interativa do protótipo de alta fidelidade no site do Figma². As telas do jogo foram feitas e um fluxo foi colocado na sequência desejada. Posteriormente, o protótipo foi avaliado por quatro pessoas, sendo três personas identificadas pelo autor Silva (2021), que podem ser vistas na Seção B, e um estudante de engenharia de *software*. Na Subseção 4.2.5.1 está detalhado o processo de validação do protótipo de alta fidelidade.

4.2.5.1 Avaliação do protótipo de alta fidelidade

Com o protótipo de alta fidelidade desenvolvido, deu-se início ao processo da validação do mesmo, que teve como objetivo avaliar o protótipo desenvolvido a partir das metas de experiência do usuário, verificar problemas e apontar soluções para melhorias no desenvolvimento do jogo. Para realizar a validação foi utilizado o processo proposto por Barbosa et al. (2021) para avaliação do protótipo.

No processo de validação do protótipo de alta fidelidade foi realizada somente uma iteração ao invés de duas iterações, como propõem Barbosa et al. (2021), devido ao pouco tempo que se tinha para executar o projeto.

Na atividade de preparação a primeira tarefa feita foi a criação do protótipo e em seguida foi definido um roteiro com uma sugestão de ordem das tarefas para o usuário seguir, ambas as tarefas podem ser vistas no Apêndice C. Logo após, foram definidos e recrutados os participantes, que foram as três personas identificadas por Silva (2021), que podem ser vistas na Seção B, e o mesmo estudante de engenharia de *software* que participou da avaliação do protótipo de baixa fidelidade; posteriormente, foi realizado a avaliação dos participantes.

Na atividade de coleta de dados os participantes executaram o teste sendo observados pelo avaliador. Logo após a realização de cada teste, o avaliador guia para a atividade de interpretação e, em seguida, é listado todos os problemas apontados pelos participantes.

Na atividade de consolidação dos resultados todos os problemas apontados foram colocados em uma tabela e na coluna ao lado dos problemas foi feita uma sugestão de correção de cada um deles por parte dos desenvolvedores. A atividade de relato dos resultados foi o fechamento desta tabela e publicação da mesma neste trabalho como pode ser vista na Tabela 6.

² <<https://www.figma.com/file/ayTQExziRpQ4my1btKOJn4/Menu-inicial?node-id=5%3A7>>

Tabela 6 – Problemas e sugestões de correção da validação do protótipo de alta fidelidade

Problema	Sugestão de correção
Jogo não ficou divertido	Adicionar mais dinamismo e animações ao jogo
Jogo sem muitos desafios	Adicionar mais fases com níveis de dificuldades diferentes

Fonte: Elaboração própria (2022)

Na Tabela 7 são apresentadas as notas dadas pelos participantes referentes às metas de experiência do jogador. As notas podem variar de um até cinco, sendo um a pior nota e cinco a melhor. Os participantes P1, P2 e P3 são as personas primária, secundária e suplementar, respectivamente, que foram definidas no trabalho de Silva (2021) e o U1 é o estudante de engenharia de *software*.

Tabela 7 – Avaliação do protótipo de alta fidelidade

Meta de Experiência	U1	P1	P2	P3	Pontuação Total	Porcentagem Total
Confiança	4	4	4	4	16	80%
Relevância	4	5	3	3	15	75%
Satisfação	5	4	3	3	15	75%
Atenção focada	3	3	4	3	13	65%
Desafio	3	4	3	3	13	65%
Diversão	3	2	3	2	10	50%

Fonte: Elaboração própria (2022)

Após a consolidação dos problemas apresentados na Tabela 6 e das notas apresentadas na Tabela 7, as metas de experiência com menor pontuação foram melhoradas no desenvolvimento do jogo. O objetivo da avaliação do protótipo de alta fidelidade de identificar problemas e apontar soluções para o jogo foi atingido. Dos problemas identificados, as sugestões de correção dos problemas levantados do jogo não ter ficado divertido e do jogo ter ficado sem muitos desafios não foram implementadas devido à falta de tempo dos autores para o desenvolvimento de mais animações e de mais fases dentro do jogo.

4.3 Documento técnico de concepção (*Technical design document*)

Nesta seção é apresentado o TDD. Corresponde a etapa do trabalho que especifica as tecnologias escolhidas para desenvolvimento, arquitetura e requisitos do jogo. O TDD é um documento que assim como o GDD, é incremental, e está em constante modificação conforme o jogo é aprimorado.

4.3.1 Tecnologias escolhidas

Para definição das tecnologias, foi feita uma pesquisa com três opções para cada tecnologia (Seção 3.7), além de ferramentas de gestão para o uso da metodologia ágil (Seção 3.2). Foram identificadas as vantagens e desvantagens de cada tecnologia pesquisada e justificativa para a escolha. Buscou-se um compartilhamento de ideias com outro Trabalho de Conclusão de Curso (TCC) de tema semelhante ao presente trabalho, confeccionado por Júnior (2021). O objetivo foi a padronização do ponto de vista técnico e organizacional, durante a produção do jogo e para trabalhos futuros. As tecnologias escolhidas estão presentes na Tabela 8.

Tabela 8 – Tecnologias escolhidas e sua justificativa

Tecnologias escolhidas	Justificativa
Docker	Ferramenta que possui um monitoramento da aplicação e os desenvolvedores já haviam conhecimento prévio.
PostgreSQL	Banco de dados que possui alta escalabilidade, suporte para dados personalizados e os desenvolvedores haviam conhecimento prévio.
Vue.js	A popularidade da comunidade, flexibilidade da ferramenta permitindo reuso de código e conhecimento prévio dos desenvolvedores.
Express.js	Facilidade na integração com banco de dados e conhecimento prévio dos desenvolvedores.
Heroku	Facilidade da ferramenta em implantações, conectividade com banco de dados PostgreSQL e preço.
Vercel	Facilidade da ferramenta construir implantações em <i>front-end</i> e preço.

Fonte: Elaboração própria (2022)

Foi necessária a utilização de uma plataforma para padronização do ambiente de desenvolvimento, a fim de reduzir esforços. A decisão do Docker justifica-se pela curva de aprendizagem da equipe, bem como o baixo custo.

A escolha do banco de dados se fez necessária para guardar informações do sistema de forma organizada. A decisão do PostgreSQL foi devido a curva de aprendizagem da equipe, flexibilidade com dados personalizados e o conhecimento prévio da tecnologia.

As escolhas de linguagens de *front-end* e *back-end* partiu do conhecimento prévio da equipe, além de uma pesquisa realizada para identificar os potenciais e fragilidades de cada linguagem (Apêndice A). Para o *back-end*, foi definido o Express.js, uma ferramenta disponível dentro do Node.js, que possui uma integração fluida com banco de dados escolhidos e escalonamento com outras plataformas e *plugins*. Para o *front-end*, a seleção do Vue ocorreu a partir de uma pesquisa de ferramentas para jogos, que eviden-

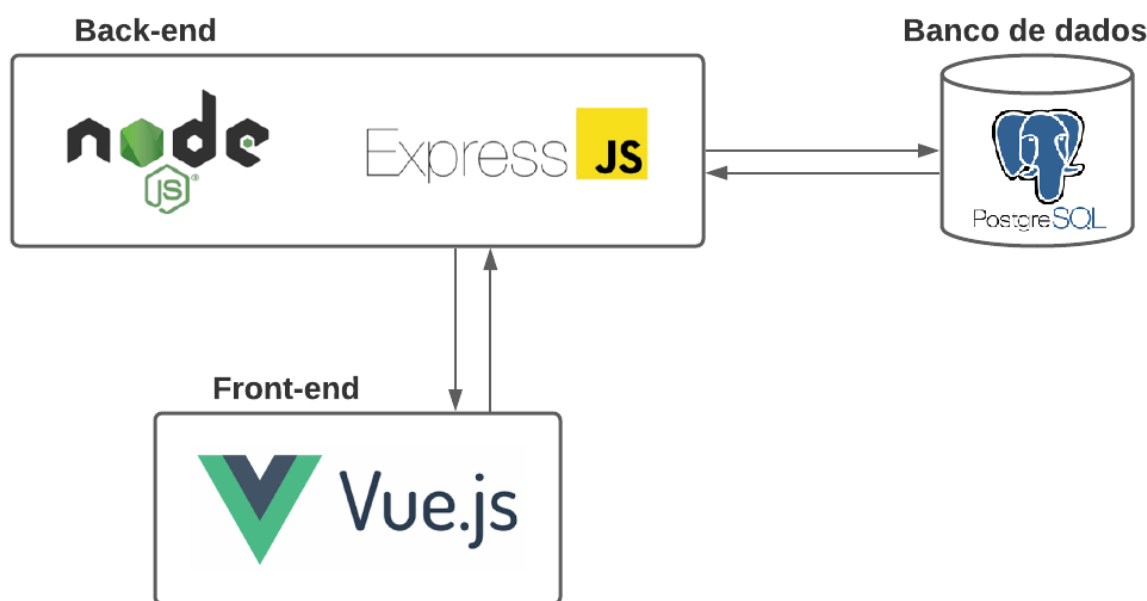
ciaram sua flexibilidade, facilidade de reutilização de componentes e linguagem de uso, tornando a curva de aprendizagem maior.

A escolha das ferramentas Heroku e Vercel para implantação do jogo partiu da estrutura do jogo. O *back-end* e *front-end* estão feitos em um único repositório, que para implantação em uma única ferramenta possui maior desafio de ser construída. Foi definido utilizar duas ferramentas, Heroku para o *back-end* e banco de dados PostgreSQL e Vercel para o *front-end*.

4.3.2 Arquitetura

A arquitetura definida para o jogo representa a comunicação e tecnologias descritas anteriormente na Seção 3.7. Na Figura 15 está a representação arquitetural do jogo.

Figura 15 – Representação arquitetural do Jogo



Fonte: Elaboração própria (2021)

As ferramentas dispostas, mostradas na Figura 15, compreendem a visualização do *software* e aspectos importantes que estão subdivididos em *back-end*, *front-end* e banco de dados.

A biblioteca ExpressJS compõe o ambiente do NodeJS, utilizado para ambientes com sistemas reduzidos e isolados, e foco em micro-serviços. A facilidade de uso das ferramentas do *back-end* com Vue auxilia no desenvolvimento do jogo, já que toda sua implementação é feita em JavaScript. A biblioteca Vue permite que o estado de elementos do jogo seja facilmente manipulado e representa uma importante vantagem para o desenvolvimento. O banco de dados relacional PostgreSQL é de fácil integração com todas as

tecnologias utilizadas e se comunica de modo direto com as tecnologias *back-end*, fazendo e recebendo requisições de dados da sessão do jogador, e do banco de perguntas e *feedback*.

A ferramenta *Docker* não está representada na Figura 15, pois é referente a fase de implantação do *software* que será descrita na próxima subseção.

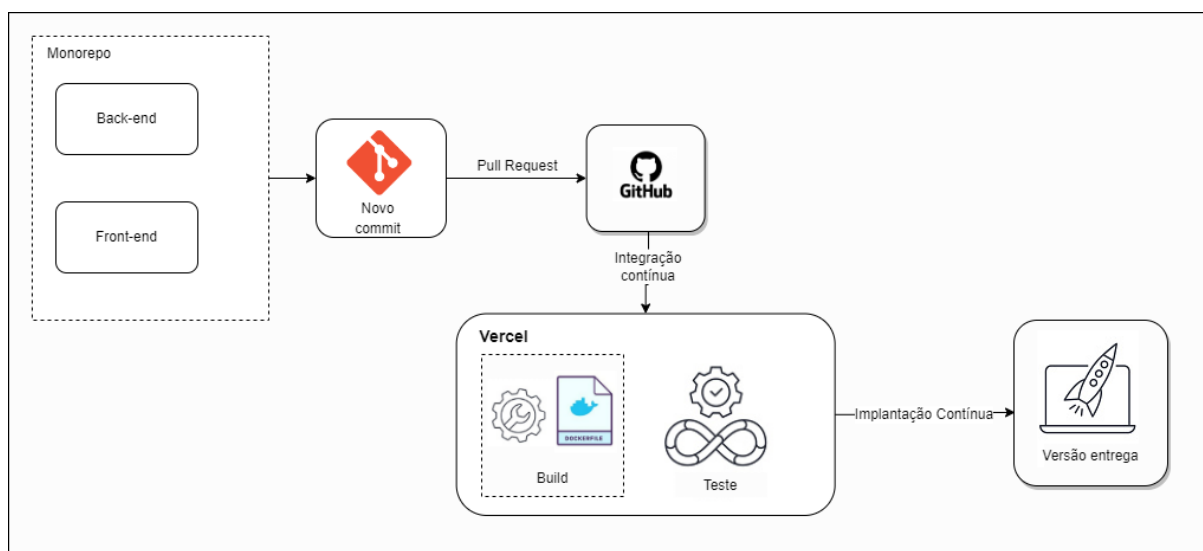
4.3.3 Integração contínua e Implantação contínua

A Integração Contínua (*continuous integration - CI*) é um método no qual o código distribuído por um desenvolvedor passa por uma série de etapas para confirmar a qualidade do código entregue.

A implantação contínua (*continuous delivery - CD*) é a entrega automática de versões de *software*, em que nenhuma ação manual é necessária para que novos recursos sejam entregues para a versão oficial de *software*. Para que essa entrega automática ocorra, o código deve primeiro passar por CI. Se o CI for implementado com sucesso, novos recursos serão lançados para os usuários finais.

Para melhor representar as etapas de CI/CD, foi definida uma *pipeline* com as etapas a serem executadas. A implantação do *back-end* e *front-end* utilizou-se a estrutura de Monorepo³, que é uma representação arquitetural que vários projetos estão em um único repositório (BROUSSE, 2019). Na Figura 16 é a representação da CI/CD do jogo.

Figura 16 – Representação da Integração contínua e Implantação contínua



Fonte: Elaboração própria (2022)

As ferramentas dispostas na Figura 16 compreendem de forma visual as etapas de integração e implantação contínua. Cada nova versão desenvolvida é enviada de forma automática para a ferramenta Vercel, que inicia a construção do sistema utilizando um

³ <<https://vercel.com/docs/concepts/git/monorepos>>

arquivo de configuração criado através da ferramenta Docker. É testada esta versão após sua construção e implantada para produção, que estará acessível para os jogadores.

4.3.4 Requisitos

Nesta subseção são apresentados os requisitos funcionais e não funcionais para o projeto. Estes recursos auxiliaram na validação e no desenvolvimento do jogo.

Na Tabela 9 estão dispostos os requisitos funcionais. Tais requisitos foram identificados a partir dos protótipos de baixa e alta fidelidade, que podem ser vistos nas Seções 4.2.4 e 4.2.5, respectivamente.

Tabela 9 – Requisitos Funcionais

ID	Épico	Descrição
RF01	Conteúdo educacional	O jogo deve apresentar conteúdo sobre heurísticas de Nielsen por meio de apresentações dos projetos e dicas nas questões.
RF02	Nível de dificuldade	O jogo deve ter dificuldades variadas para as questões, referentes ao avanço do jogador e a cada projeto escolhido.
RF03	<i>Feedback</i> para respostas	O jogador deve receber uma mensagem após acertar ou errar alguma questão. Caso a resposta esteja incorreta, o jogo deve exibir a resposta correta com uma explicação. Caso a resposta esteja correta o jogo deve mostrar uma mensagem parabenizando.
RF04	<i>Feedback</i> para respostas	A cada <i>feedback</i> referente as questões acertadas devem ser apresentada na tela a pontuação.
RF05	Pontuação do jogador	O jogo deve apresentar a pontuação atual do jogador e o número de questões acertadas/questões totais.
RF06	Fluxo da partida	O jogo deve permitir o jogador voltar ao menu principal, <i>ranking</i> e menu do jogo.
RF07	Fluxo da partida	O jogo deve bloquear os níveis ainda não atingidos, sendo possível alcançá-los após uma pontuação determinada para cada nível.
RF08	Fluxo da partida	O jogo deve instruir o jogador a próxima fase.
RF09	Ranking	O jogo deve permitir o jogador acessar o ranking e apresentá-lo por pontuação e porcentagem de respostas corretas.
RF10	Explicação do jogo	O jogo deve fazer a apresentação das regras, contexto e demais elementos para o usuário, assim que iniciada uma nova partida.
RF11	Seleção de personagem	O jogo deve permitir o jogador escolher seu personagem no início do jogo.
RF12	Fases dinâmicas	O jogo deve trazer perguntas diferentes e em ordem randômica em cada uma das fases e a cada novo início de jogo.

Fonte: Elaboração própria (2021)

Os requisitos funcionais compreendem todas as necessidades e problemas que foram atendidas e previstos pelo *software*. Os requisitos identificados definiram as tarefas a serem desenvolvidas no jogo para que não houvesse erros de interpretação no planejamento do desenvolvimento. Os requisitos RF10, RN11 e RF12 representam as melhorias apontadas nos jogos pesquisados com ênfase em IHC e nas heurísticas de Nielsen que estão presentes na Tabela 1.

Na Tabela 10 são apresentados os requisitos não funcionais levantados para o projeto. Estes auxiliaram no comportamento do *software* e qualquer elemento externo ao jogo. Eles foram identificados a partir dos documentos GDD e TDD (Seções 4.2 e 4.3).

Tabela 10 – Requisitos Não-Funcionais

ID	Épico	Descrição
RNF01	Eficiência	O jogo deve processar as requisições do jogador em até três segundos.
RNF02	Confiabilidade	O jogo deve permanecer no ar por pelo menos 99% do tempo.
RNF03	Portabilidade	O jogo deverá ser executado em navegadores Chrome, Opera e Mozilla Firefox
RNF04	Implementação	O jogo deve ser desenvolvido totalmente em Javascript para melhor manutenção e evolução.

Fonte: Elaboração própria (2021)

Os requisitos não funcionais (RNFs) são requisitos que declaram restrições, ou atributos de qualidade para um *software* e/ou para o processo de desenvolvimento deste sistema (CYSNEIROS; LEITE, 2001). Para identificá-los, foram analisados os aspectos do *software*.

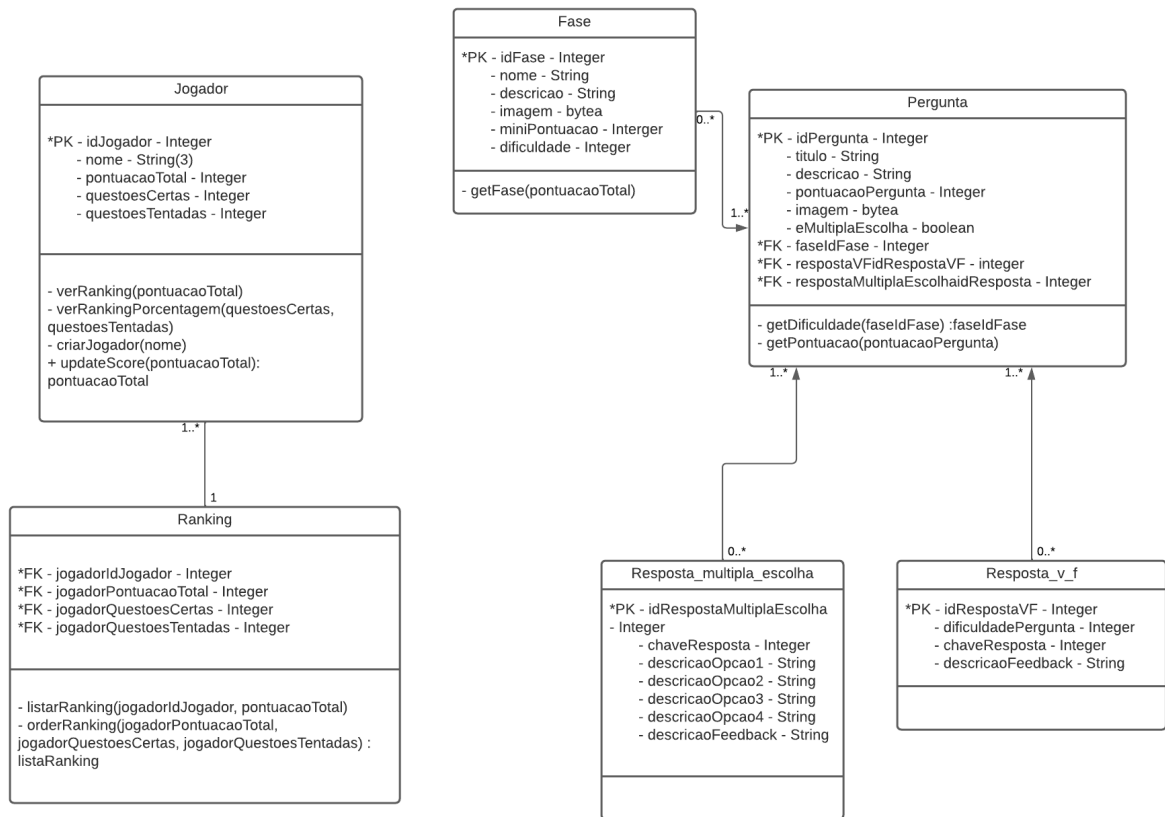
4.3.5 Diagramas

Nesta subseção são apresentados os diagramas de Classe, Atividades e de Pacotes que representam uma visão arquitetural de diferentes aspectos do jogo.

4.3.5.1 Diagrama de Classes

Os diagramas de classes são utilizados para ilustrar de forma objetiva a estrutura do sistema, por meio de classes, atributos e relações entre seus objetos. Na Figura 17 é apresentado o diagrama de classes do jogo com objetos e suas relações.

Figura 17 – Diagrama de Classes



Fonte: Elaboração própria (2022)

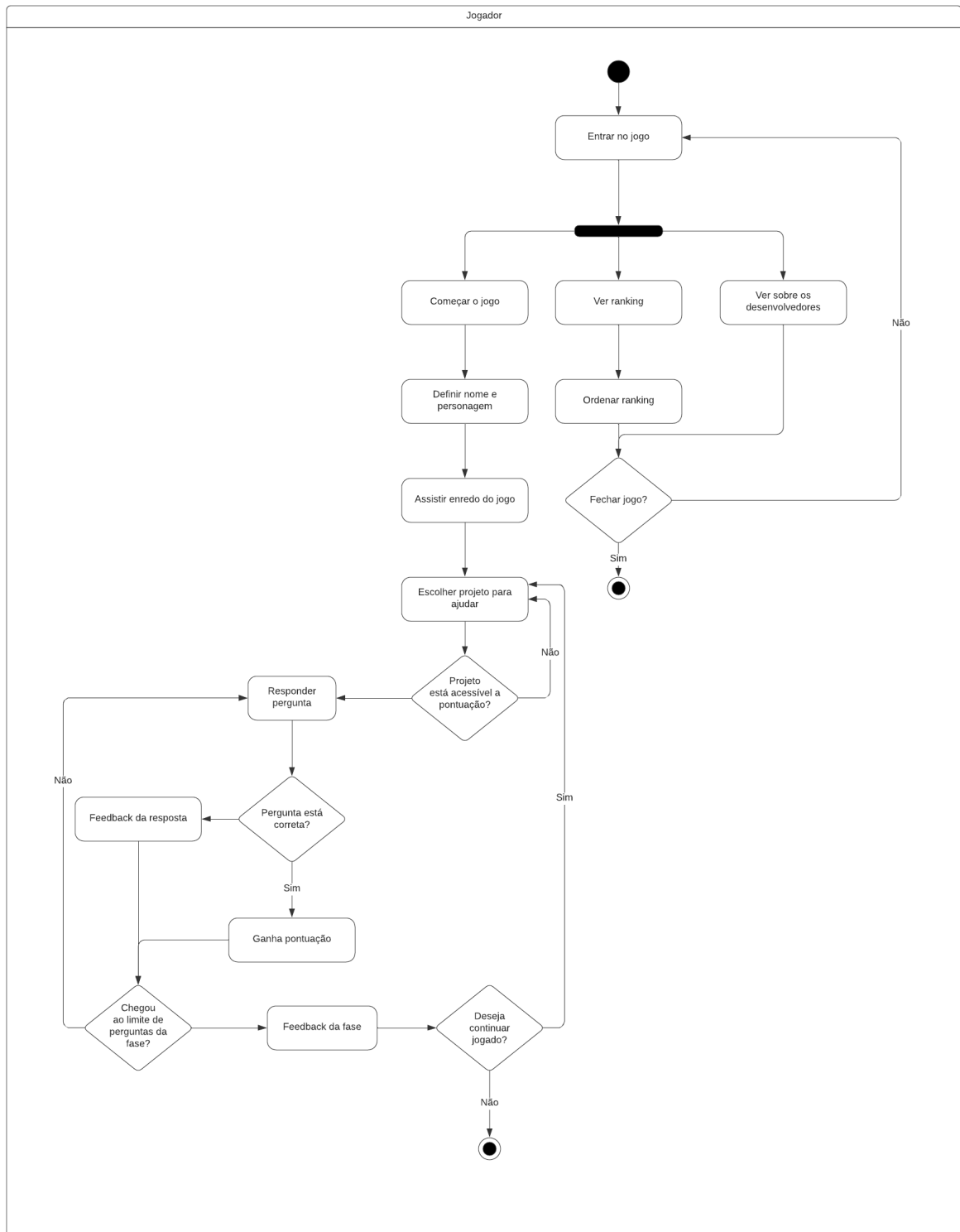
Como mostrado na Figura 17, as classes criadas para o desenvolvimento do jogo representam os modelos desenvolvidos para essas classes. Nelas, têm-se o modelo de jogador que conecta ao *Ranking*, a partir da pontuação total e quantidade de questões. Para o outro modelo, existem conexões entre a classe Fase, Pergunta e os tipos de perguntas.

4.3.5.2 Diagrama de Atividades

Os diagramas de atividade são utilizados para ilustrar o fluxo de um sistema e mapear os passos envolvidos na execução de uma atividade, levando em consideração a sequência e as condições do fluxo da atividade.

Na Figura 18 é apresentado o diagrama de atividades com o fluxo do jogo e as possíveis decisões do jogador.

Figura 18 – Diagrama de atividades



Fonte: Elaboração própria (2022)

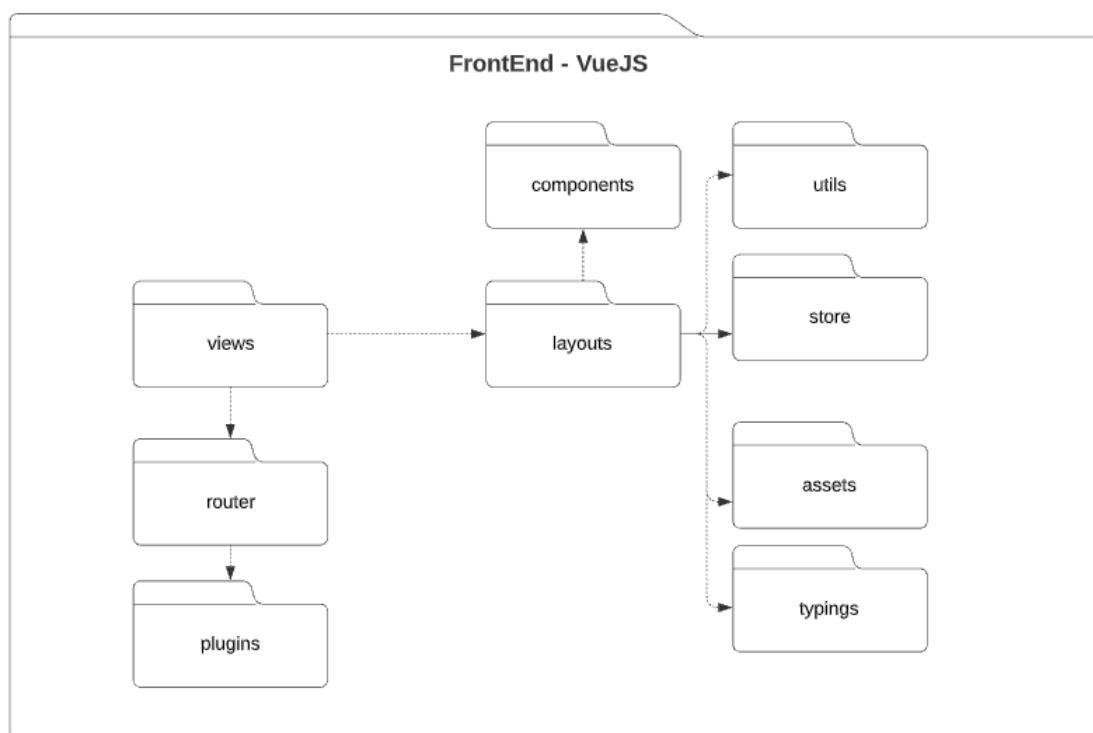
Como mostrado na Figura 18, o fluxo descrito no diagrama representa os possíveis caminhos que um jogador pode seguir para completar o jogo, desde o início, percorrendo o fluxo das perguntas, até a finalização das fases e ordenação do *ranking*.

4.3.5.3 Diagrama de Pacotes

Os diagramas de pacotes são utilizados para mostrar, em forma de pacotes, como é a disposição e organização dos elementos do modelo. Cada elemento é colocado de forma hierárquica para ilustrar a organização de classe, diagrama, pasta ou pacote.

Na Figura 19 é apresentado o diagrama de pacotes para o subsistema de *front-end* que compõe os elementos do jogo, dividido em sistemas e classes de interação.

Figura 19 – Diagrama de pacotes representação *front-end*



Fonte: Elaboração própria (2022)

Na Figura 19 é apresentado os oito elementos presentes no *front-end* do jogo: *components*, *layout*, *utils*, *store*, *assets*, *typping*, *views*, *router* e *plugging*s.

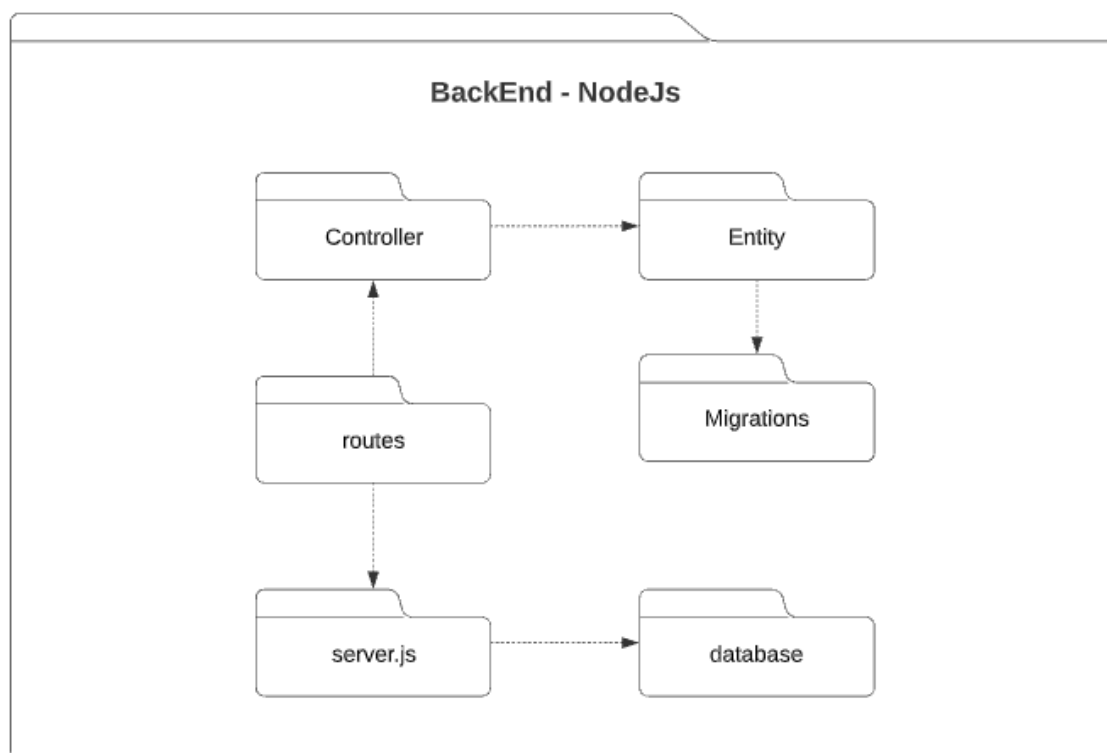
O **components** possui os componentes utilizados nas telas, como botões, caixas de texto, modais, etc. O **layout** possui o tamanho dos elementos da página. O **utils** possui exportações de métodos utilizados, como por exemplo método *random* de coletar perguntas. No pacote **store** armazena-se as ações salvas pelo usuário que não necessitam ser salvas em banco de dados. O **assets** possui a definição de *cores*, fontes e imagens utilizadas.

O **typping** possui os tipos de dados para o pacote de *store*. A **views** contém todas as páginas do jogo. As páginas utilizam os pacotes *components*, *layouts* e *utils* do sistema.

O **router** possui entradas e saídas do sistema, indicando para próximo fluxo do jogo. O **pluggings** possui plugins utilizados para fonte do texto e servidor.

Na Figura 20 é apresentado o diagrama de pacotes para o subsistema de *back-end* que compõe os elementos do jogo, dividido em sistemas e classes de interação.

Figura 20 – Diagrama de pacotes representação *back-end*



Fonte: Elaboração própria (2022)

Na Figura 20 é apresentado os oito elementos presentes no *back-end* do jogo: *controller*, *entity*, *migrations*, *routes*, *server.js* e *database*.

A **controller** possui a implementação das rotas definidas em *routes*. O **entity** possui as definições de tabelas do banco de dados. A **migrations** possui o versionamento dos *schema* da aplicação. O **routes** é o módulo responsável pelas saídas e entradas de serviços definindo seus *endpoints*. O **server.js** é o arquivo que inicializa toda a aplicação. O **database** é o módulo que possui as configurações do banco de dados.

No próximo capítulo são apresentadas as telas do jogo Universidade da Heurística e o fluxo da partida.

5 Universidade das Heurísticas

Nesta seção são apresentadas as telas do jogo para ensino-aprendizagem das heurísticas de Nielsen, denominado Universidade das Heurísticas. O jogo pode ser acessado no site criado para ele, o Universidade das Heurísticas¹.

Para a estrutura do jogo e o fluxo de funcionalidades, foi criado um protótipo de papel (Apêndice B). Após isso, foi confeccionado o protótipo de alta fidelidade (Apêndice C) que compõe o *design* e um refinamento do fluxo do jogo. Posteriormente, esse protótipo foi validado e a partir dele, levantados os requisitos funcionais e não funcionais do jogo. A seguir são apresentadas as telas do jogo.

A Figura 21 representa a tela inicial. Nesta tela é apresentado o primeiro contato com o jogo e possui apenas um botão “Começar”, que redireciona o jogador para a tela de Menu (Figura 22).

Figura 21 – Tela Inicial do jogo

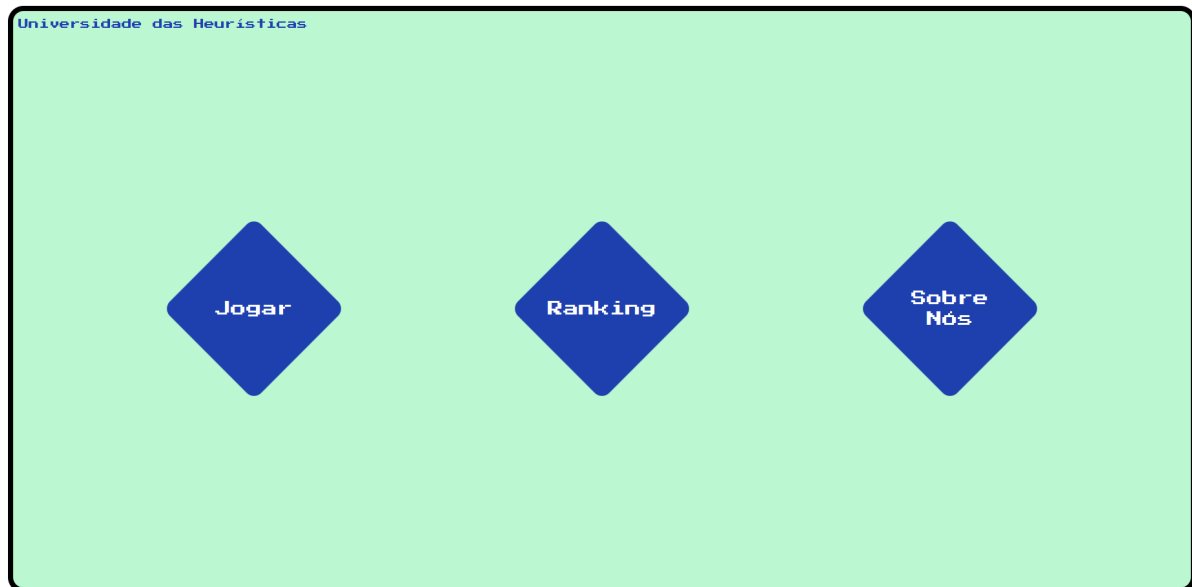


Fonte: Elaboração própria (2022)

Na Figura 22 pode-se visualizar três opções: **Jogar**, ambiente que o jogador pode escolher as fases, responder as perguntas e entender como jogar; **Ranking**, ambiente que são listados todos os jogadores e suas respectivas pontuações e **Sobre nós**, espaço que é apresentado o contexto do jogo, como contribuir e agradecimentos.

¹ <<https://universidade-das-heurísticas.vercel.app>>

Figura 22 – Menu Inicial do jogo



Fonte: Elaboração própria (2022)

Após clicar em jogar, o jogador escolhe o seu nome, com até três caracteres, seu personagem, então, é redirecionado para uma tela de boas-vindas, como mostrado na Figura 23, com resumo do jogo e uma imagem da Universidade de Brasília (UnB) em baixa resolução afim de simular o efeito em 8 bits. A escolha de personagem foi implementada e está presente na Tabela 1 como uma das melhorias que foram implementadas.

Figura 23 – Tela de boas vindas do jogo

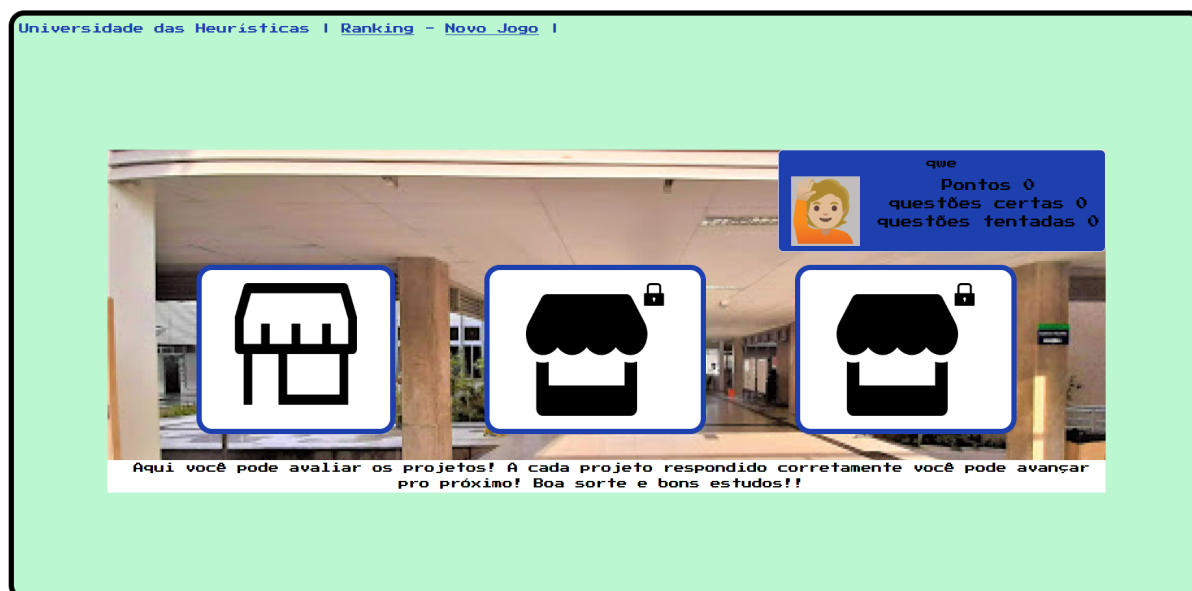


Fonte: Elaboração própria (2022)

Ao avançar a tela de boas-vindas, são apresentadas as fases do jogo, como mostrado

na Figura 24. Nesta tela são apresentadas três fases com dificuldades diferentes e para acessar as fases bloqueadas é necessário alcançar trinta e sessenta pontos, respectivamente.

Figura 24 – Tela das fases do jogo

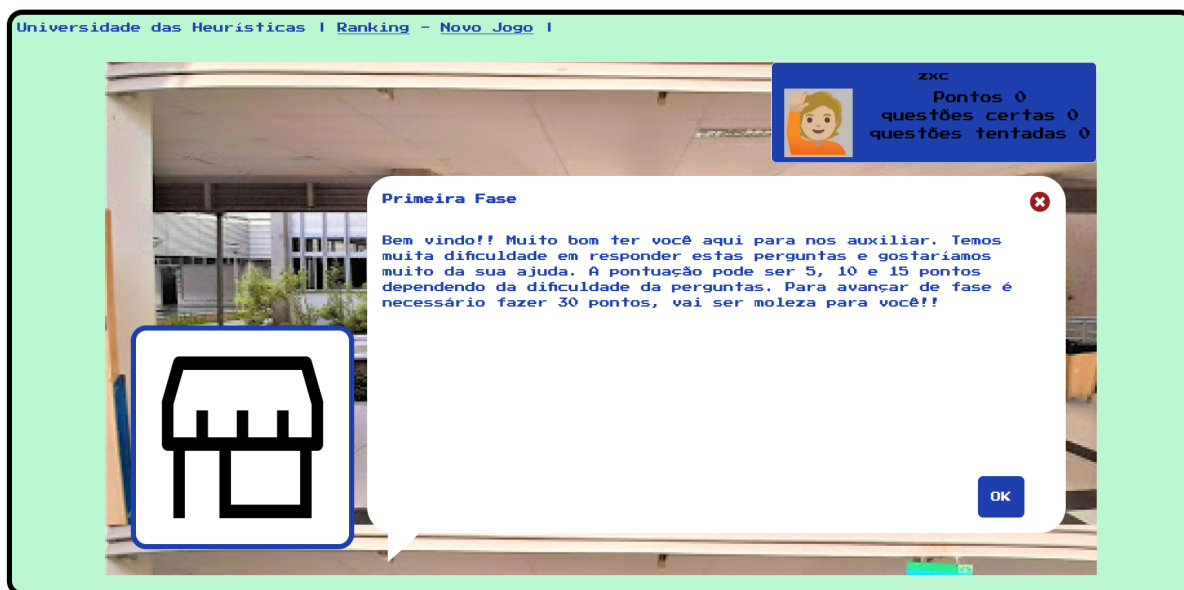


Fonte: Elaboração própria (2022)

Como pode ser visto na Figura 24, as duas fases com maior nível de dificuldade não foram desenvolvidas, mas incentiva-se que sejam criadas em trabalhos futuros. Nesta tela é possível ver o nome e o personagem escolhido pelo jogador, total de questões acertadas e tentadas, além de *links* para *Ranking* e Novo Jogo.

Ao selecionar uma fase, o jogador é redirecionado para sua descrição, como mostrado na Figura 25. Nesta tela é apresentado um resumo, pontuação das perguntas e pontuação mínima para avançar de fase. Nesta tela foi implementado a explicação do jogo que está presente na Tabela 1 como uma das melhorias que foram implementadas.

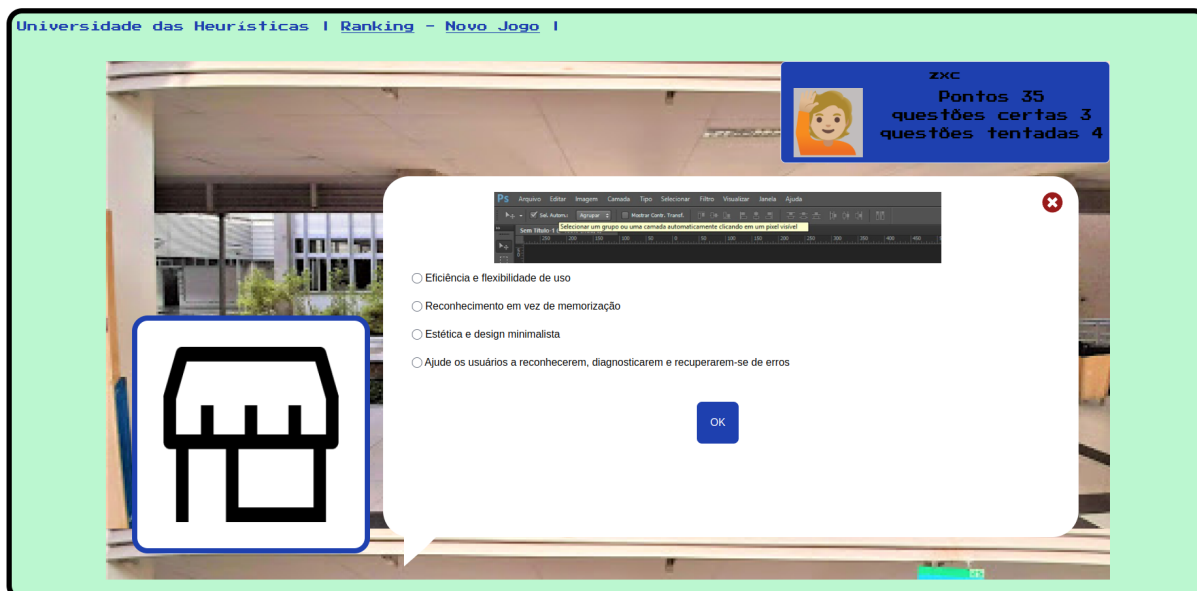
Figura 25 – Descrição da primeira fase



Fonte: Elaboração própria (2022)

Na próxima tela são apresentadas as perguntas, divididas em múltipla escolha, como mostrado na Figura 26, e verdadeiro ou falsas, mostrado na Figura 27.

Figura 26 – Pergunta de múltipla escolha



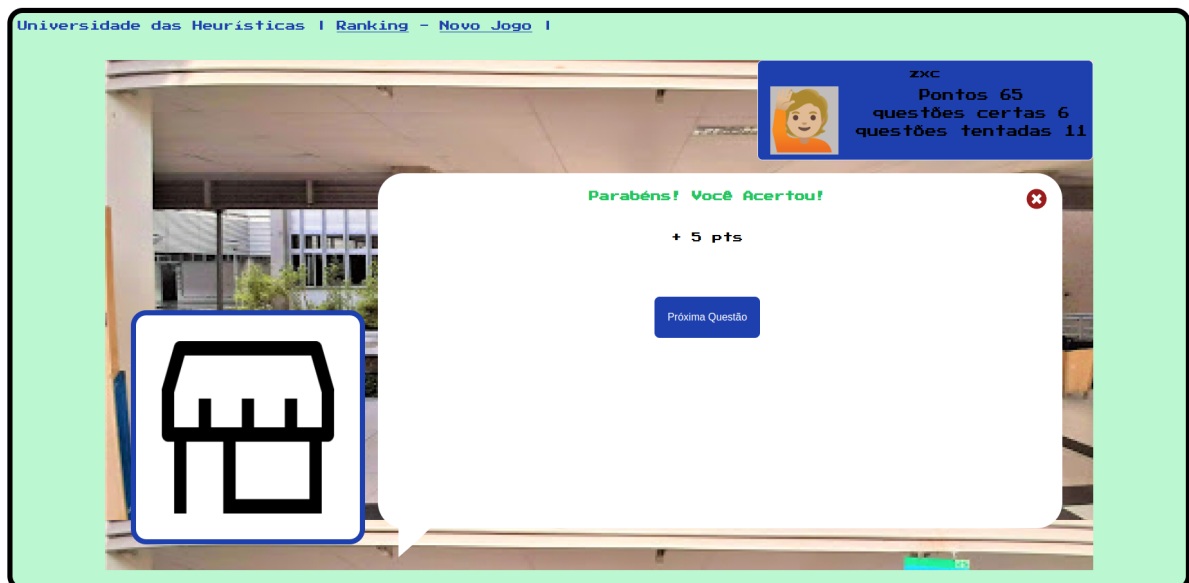
Fonte: Elaboração própria (2022)

Figura 27 – Pergunta de verdadeiro ou falso

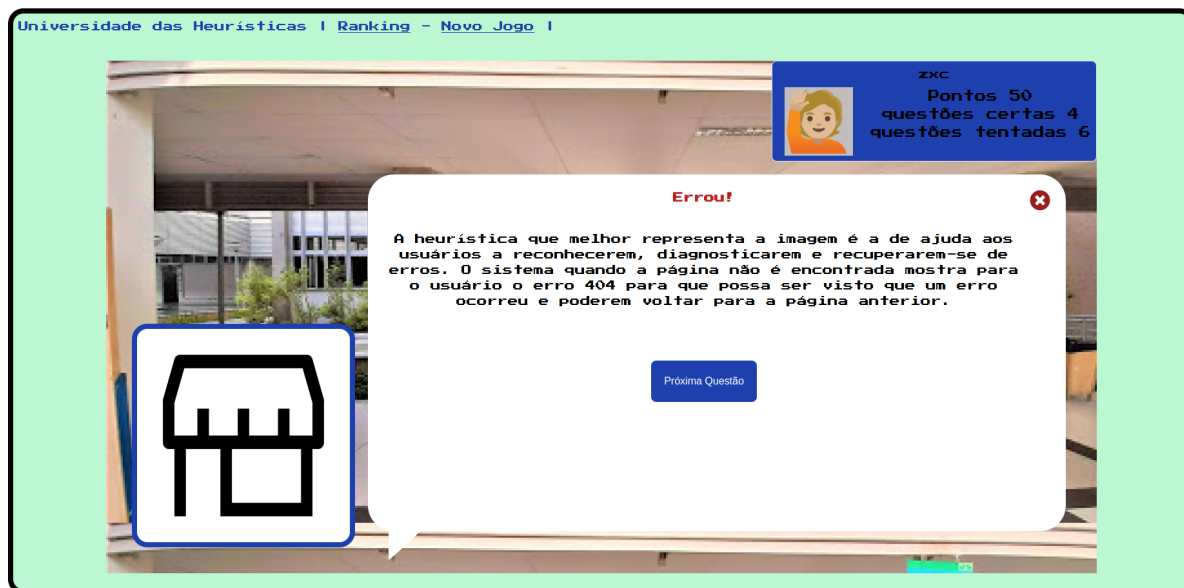


Fonte: Elaboração própria (2022)

Após responder uma pergunta, o *feedback* é exibido na tela para o jogador. Caso a resposta esteja correta, é apresentada a pontuação adquirida, conforme a Figura 28. Caso a resposta esteja errada, é apresentada a resposta correta e uma explicação do por quê, conforme mostrado na Figura 29.

Figura 28 – *Feedback* resposta correta

Fonte: Elaboração própria (2022)

Figura 29 – *Feedback* resposta incorreta

Fonte: Elaboração própria (2022)

Nessas telas de perguntas foi implementado as fases dinâmicas que está presente na Tabela 1 como uma das melhorias que foram implementadas. Todas as telas descritas no Menu jogar possuem um *link* para tela *Ranking* e Novo jogo. Na tela de *Ranking* é possível ordenar por pontuação total, porcentagem de acerto e nome escolhido pelos jogadores, em ordem alfabética, conforme mostrado na Figura 30.

Figura 30 – Tela de *Ranking*

The screenshot shows a ranking screen with a green header bar containing the text "Universidade das Heurísticas - Ranking" and a "Voltar" button. Below the header is a table with three columns: "Nome", "Pontos", and "Questões ↓". The table contains three rows of data:

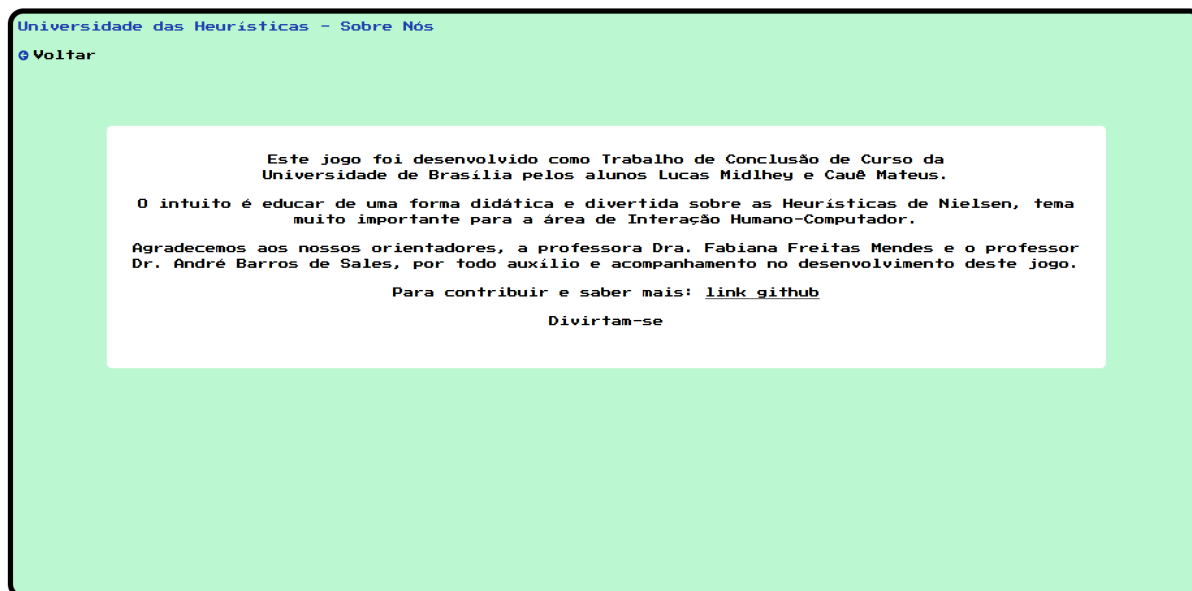
Nome	Pontos	Questões ↓
zxc	65	54.55%
xxx	265	53.57%
que	90	47.37%

Fonte: Elaboração própria (2022)

A tela Sobre nós, mostrada na Figura 31, possui informações sobre o intuito do

trabalho, os desenvolvedores, agradecimentos e como contribuir com o jogo, por meio de um link do *GitHub*.

Figura 31 – Tela do jogo com informações adicionais



Fonte: Elaboração própria (2022)

O jogador pode continuar respondendo as perguntas do jogo para alcançar melhor pontuação. O jogo atualmente possui 60 perguntas, como descrito na Seção 4.2.3.2, sendo capaz de criar mais como trabalho futuro. No próximo capítulo será apresentado a conclusão deste trabalho, trabalhos futuros e evolução dos autores.

6 Conclusão

Este projeto se propôs a desenvolver um jogo digital que auxilie o processo de ensino-aprendizagem de alunos, envolvendo conceitos sobre as heurísticas de Nielsen em interação humano-computador. O objetivo geral de criar um jogo digital com foco no auxílio do ensino-aprendizagem de heurísticas de Nielsen foi cumprido.

Todos os objetivos específicos: (1) Aplicar os conceitos relacionados a jogos digitais de apoio ao processo de ensino-aprendizagem; (2) Investigar as heurísticas de Nielsen; (3) Desenvolver um jogo para ensino-aprendizagem; (4) Utilizar boas práticas de programação e processo de desenvolvimento de *software* no desenvolvimento do jogo; (5) Utilizar os conceitos das heurísticas de usabilidade no jogo; e (6) Desenvolver melhorias identificadas nos jogos *UsabiliCity*, *UsabilityGame* e *MACteaching*, também foram cumpridos. As melhorias propostas em comparação aos jogos pesquisados, que podem ser vistas na Tabela 1, também foram implementadas.

As metodologias empregadas neste trabalho foram a aplicação prática do *Play-centric Design Process*, *Scrum* e *Kanban*. Todos os elementos conceituais que basearam o desenvolvimento e execução do jogo foram estudados e colocados em prática neste trabalho. Quanto ao processo de idealização, planejamento, ciclo de vida do jogo e resultado do desenvolvimento foram apresentados ao longo do trabalho por meio da exposição escrita e por figuras.

O Universidade das Heurísticas pode auxiliar o ensino-aprendizagem das heurísticas de Nielsen, empregando a prática em um jogo digital, de forma descontraída. O projeto é *open source* e foi utilizado o repositório Universidade-das-Heurísticas¹. Além disso, o jogo pode ser acessado via link².

Embora os resultados alcançados mostram relevância de jogos no processo de ensino-aprendizagem, listamos pontos relevantes que contribuirão em pesquisas futuras, são eles: (1) Estudos envolvendo a aplicação do jogo e a análise de dados; (2) Desenvolvimento de um jogo ou uma plataforma capaz de unir outros trabalhos desenvolvidos na área de IHC; (3) Expansão do jogo Universidade das Heurísticas com a criação de novas fases, níveis, perguntas e novos elementos; (4) Criação de novos modos no jogo; (5) e Melhorias no *design*; (6) Ajustar e revisar as perguntas com testes afim de deixá-las mais claras; (7) Avaliar o jogo através de testes com usuários; (8) Iniciar a fase de garantia da qualidade vista na Subseção 2.4. Esta lista entrega possíveis melhorias ao jogo, que trarão avanços no processo de ensino-aprendizagem utilizando o Universidade das Heurísticas,

¹ <<https://github.com/RecursosDigitaisdeEnsinoAprendizagemIHC/Universidade-das-Heurísticas>>

² <<https://universidade-das-heurísticas.vercel.app>>

disponibilizando um jogo com excelência no que ele se propõe.

A pandemia da COVID-19 mudou a forma como o ensino é transmitido para os alunos. Docentes e discentes não estavam preparados para as rápidas transformações que o momento impôs, mas têm se adaptado de acordo com as ferramentas disponíveis, que auxiliam no processo. Com a retomada das universidades de forma presencial, algumas dessas ferramentas deixarão de ser usadas e outras permanecerão auxiliando no processo de ensino-aprendizagem, mesmo no ensino presencial. Nesse contexto, o Universidade das Heurísticas foi desenvolvido para auxiliar esse processo de ensino-aprendizagem, tanto de forma remota quanto presencial, com foco nas heurísticas de Nielsen.

Para tanto, o trabalho realizado neste jogo contribuiu para o desenvolvimento de ambos os autores no quesito programação. Não tínhamos o costume de desenvolver *softwares* no estilo *full stack*, pois um de nós estava acostumado a trabalhar só com banco de dados e manutenção de sistemas legado, e o outro autor, estava a muitos anos sem programar algo que não fosse em *Visual Basic Advanced* (VBA) ou *Data Analysis Expressions* (DAX). Além dessas experiências dos autores, existe o fato de que o processo de construção de um jogo difere da construção de uma aplicação tradicional, pois em um jogo além de levar em consideração os conhecimentos aprendidos durante o curso de Engenharia de *Software*, tivemos que trabalhar com a metodologia do *Playcentric Design Process* que traz uma abordagem diferente que é centrada na experiência do jogador.

Referências

- AGUIAR, M. B. N. d. et al. Desenvolvimento de softwares para o ecossistema do turismo brasileiro. Florianópolis, SC., 2020. Citado na página 45.
- ALEXANDRE, T. M. de; JUNIOR, J. R. de A. Gestão do desenvolvimento de software com o uso de quadro virtual kanban. *Brazilian Journal of Development*, v. 6, n. 12, p. 103726–103749, 2020. Citado na página 37.
- ALMEIDA, R. S. e G. Abordagem híbrida na gestão de projetos de tecnologia e automatização de processos. *Boletim do Gerenciamento*, v. 19, n. 19, p. 20–32, 2020. ISSN 2595-6531. Disponível em: <<https://nppg.org.br/revistas/boletimdogerenciamento/article/view/470>>. Citado na página 20.
- BARBOSA, S. D. J. et al. *Interação Humano-Computador e Experiência do Usuário*. [S.l.]: Autopublicação, 2021. Citado 5 vezes nas páginas 17, 39, 51, 52 e 54.
- BOEG, J. Kanban em 10 passos. *Tradução de Leonardo Campos, Marcelo Costa, Lúcio Camilo, Rafael Buzon, Paulo Rebelo, Eric Fer, Ivo La Puma, Leonardo Galvão, Thiago Vespa, Manoel Pimentel e Daniel Wildt. C4Media*, 2010. Citado na página 37.
- BORGES, K. S.; RAPKIEWICZ, C. E.; FEIJÓ, A. C. Usando heurísticas de nielsen para avaliar objetos de aprendizagem e softwares educacionais: um estudo exploratório na área de matemática para ensino superior. In: *Anais do Workshop de Informática na Escola*. [S.l.: s.n.], 2012. v. 1, n. 1. Citado 2 vezes nas páginas 40 e 41.
- BORTOLUCI, R. et al. Análise da aplicabilidade das propriedades do kanban nos modelos de desenvolvimento de software. *Revista Científica on-line-Tecnologia, Gestão e Humanismo*, v. 7, n. 2, 2017. Citado na página 37.
- BRITO, R. et al. Macteaching: Utilizando um jogo para apoio ao ensino do método de avaliação de comunicabilidade. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, v. 27, n. 1, p. 796, 2016. ISSN 2316-6533. Disponível em: <<https://www.br-ie.org/pub/index.php/sbie/article/view/6765>>. Citado 2 vezes nas páginas 18 e 43.
- BROUSSE, N. The issue of monorepo and polyrepo in large enterprises. In: *Proceedings of the Conference Companion of the 3rd International Conference on Art, Science, and Engineering of Programming*. [S.l.: s.n.], 2019. p. 1–4. Citado na página 58.
- Burak Karakan. *What exactly is Docker?* 2020. <<https://medium.com/swlh/what-exactly-is-docker-1dd62e1fde38#:~:text=They%20allow%20running%20multiple%20workloads,are%20pretty%20lightweight%20and%20efficient.>> Accessed: 2022-03-09. Citado na página 80.
- CHANDLER, H. M. *Manual de produção de jogos digitais*. [S.l.]: Bookman Editora, 2009. Citado na página 35.

- CLÍMACO, G. d. S. Aplicação do jogo usability na disciplina interação humano-computador: um estudo comparativo. p. 11, 2016. Citado na página 18.
- CYSNEIROS, L. M.; LEITE, J. Requisitos não funcionais: da elicitação ao modelo conceitual. *PhDTese, PUC-RJ*, 2001. Citado na página 60.
- DÖRNER, R. et al. *Serious Games*. [S.l.]: Springer, 2016. Citado 2 vezes nas páginas 42 e 43.
- Editorial Aela. *Heurísticas De Nielsen: 10 Dicas Para Melhorar a Usabilidade De Sua Interface*. 2019. <<https://aelaschool.com/experienciadousuario/10-heurísticas-nielsen/>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.
- EVANGELISTA, C. S. J. Desenvolvimento de jogo multiplayer para plataforma web utilizando php e javascript. 2013. Citado na página 44.
- FALKEMBACH, G. A. M.; GELLER, M.; SILVEIRA, S. R. Desenvolvimento de jogos educativos digitais utilizando a ferramenta de autoria multimídia: um estudo de caso com o toolbook instructor. *RENOTE*, v. 4, n. 1, Jul 2006. ISSN 1679-1916. Disponível em: <<https://seer.ufrgs.br/renote/article/view/13874>>. Citado na página 51.
- FERNANDES, K. T.; LUCENA, M. J. N. R.; ARANHA, E. H. da S. Uma experiência na criação de game design de jogos digitais educativos a partir do design thinking. *RENOTE*, v. 16, n. 1, 2018. Citado na página 34.
- FERREIRA, B. M. et al. Usability: Um jogo de apoio ao ensino de propriedades de usabilidade de software através de analogias. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, v. 25, n. 1, p. 1273, 2014. ISSN 2316-6533. Disponível em: <<https://br-ie.org/pub/index.php/sbie/article/view/3075>>. Citado 2 vezes nas páginas 18 e 43.
- FERREIRA, D. M. P. *Análise e comparação de Desempenho de Containers em Docker e em Apache Mesos*. Tese (Doutorado), 2021. Citado na página 44.
- FORLIZZI, J.; BATTARBEE, K. Understanding experience in interactive systems. In: . New York, NY, USA: Association for Computing Machinery, 2004. (DIS '04), p. 261–268. ISBN 1581137877. Disponível em: <<https://doi.org/10.1145/1013115.1013152>>. Citado na página 39.
- Fullerton, T. Play-centric games education. *Computer*, v. 39, n. 6, p. 36–42, 2006. Citado na página 33.
- FULLERTON, T. *Game design workshop: a playcentric approach to creating innovative games*. [S.l.]: CRC press, 2014. Citado 5 vezes nas páginas 27, 33, 34, 35 e 42.
- Gabriel Moma. *10 heurísticas de Nielsen para o design de interface*. 2017. <<https://brasil.uxdesign.cc/10-heurísticas-de-nielsen-para-o-design-de-interface-58d782821840>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.
- Gabrielle Pinhata. *10 heurísticas de Nielsen: Melhorando a usabilidade de suas interfaces*. 2021. <<https://www.programmers.com.br/blog/10-heurísticas-de-nielsen-melhorando-a-usabilidade/>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.

- GIL, A. C. Como elaborar projetos de pesquisa/antonio carlos gil.—10. reimpr. *São Paulo*, 2002. Citado na página 18.
- GRENNING, J. Planning poker or how to avoid analysis paralysis while release planning. *Hawthorn Woods: Renaissance Software Consulting*, v. 3, p. 22–23, 2002. Citado na página 49.
- HARZING, A.-W.; ALAKANGAS, S. Google scholar, scopus and the web of science: A longitudinal and cross-disciplinary comparison. *Scientometrics*, v. 106, p. 787–804, 02 2016. Citado na página 29.
- HASSENZAHN, M. *Experience Design: Technology for All the Right Reasons*. Morgan & Claypool, 2010. (Synthesis lectures on human-centered informatics). ISBN 9781608450473. Disponível em: <<https://books.google.com.br/books?id=pwGsZZBBILAC>>. Citado na página 39.
- HASSENZAHN, M.; TRACTINSKY, N. User experience - a research agenda. *Behaviour and Information Technology*, v. 25, p. 91 – 97, 03 2006. Citado na página 39.
- Henrique Carvalho. *10 Heurísticas de Nielsen para avaliar a interface*. 2020. <<https://vidadeproduto.com.br/heuristicas-de-nielsen/>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.
- HEWETT, T. T. et al. *ACM SIGCHI Curricula for Human-Computer Interaction*. New York, NY, USA, 1992. Citado na página 17.
- Hussain Arif. *Use Express.JS as a Web Server*. 2020. <<https://medium.com/easyread/introduction-to-express-js-246191ec05f2>>. Accessed: 2022-03-09. Citado na página 81.
- ISO-9241-210:2019. *Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems*. [S.l.], 2019. v. 2019. Citado na página 39.
- JúNIOR, R. R. P. Personadesiggame (pdg): Um jogo educacional sobre personas. p. 91, 2021. Citado 3 vezes nas páginas 43, 51 e 56.
- KAMMERSGAARD, J. Four different perspectives on human–computer interaction. *International Journal of Man-Machine Studies*, v. 28, n. 4, p. 343–362, 1988. ISSN 0020-7373. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0020737388800178>>. Citado na página 39.
- KAUARK, F. d. S. et al. Metodologia da pesquisa: um guia prático. Via Litterarum, 2010. Citado 2 vezes nas páginas 24 e 25.
- Lucio Chiessi. *Why should I use PostgreSQL as Database in my Startup/Company*. 2018. <<https://medium.com/we-build-state-of-the-art-software-creating/why-should-i-use-postgresql-as-database-in-my-startup-company-96de2fd375a9>>. Accessed: 2022-03-09. Citado na página 80.
- MACIEL, C. et al. Avaliação heurística de sítios na web. *VII ESCOLA DE INFORMÁTICA DO SBC-CENTROOESTE*, 2004. Citado na página 41.
- MACRAE, C. *Vue.js: Up and Running: Building Accessible and Performant Web Apps*. [S.l.]: "O'Reilly Media, Inc.", 2018. Citado 2 vezes nas páginas 44 e 45.

- MAGALHÃES, L. B. et al. Análise comparativa dos algoritmos de otimização de consultas do postgresql. In: *Colloquium Exactarum. ISSN: 2178-8332*. [S.l.: s.n.], 2015. v. 7, n. 1, p. 01–21. Citado na página 44.
- MAHNIC, V. Improving software development through combination of scrum and kanban. *Recent Advances in Computer Engineering, Communications and Information Technology, Espanha*, p. 281–288, 2014. Citado na página 38.
- Marco Bruno. *10 Heurísticas de Nielsen. Uma fórmula pra evitar erros básicos de usabilidade*. 2019. <<https://www.alura.com.br/artigos/10-heuristicas-de-nielsen-uma-formula-para-evitar-erros-basicos-de-usabilidade>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.
- MARDAN, A. Using express.js to create node.js web apps. In: *Practical Node.js*. [S.l.]: Springer, 2018. p. 51–87. Citado na página 44.
- MARQUES, V. C.; AMARAL, S. F. do. As necessidades educacionais evidenciadas pela pandemia de covid-19. *Internet Latent Corpus Journal*, v. 10, n. 1, p. 6–19, 2020. Citado na página 17.
- MORAN, T. P. The command language grammar: a representation for the user interface of interactive computer systems. *International Journal of Man-Machine Studies*, v. 15, n. 1, p. 3–50, 1981. ISSN 0020-7373. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0020737381800223>>. Citado na página 39.
- NETO, A. S. Desenvolvimento de uma aplicação pwa que comporte outras aplicações usando arquitetura de micro frontend. Pontifícia Universidade Católica de Goiás, 2020. Citado na página 45.
- NIELSEN, J. Enhancing the explanatory power of usability heuristics. In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. [S.l.: s.n.], 1994. p. 152–158. Citado na página 40.
- NIELSEN, J. Usability inspection methods. In: *Conference companion on Human factors in computing systems*. [S.l.: s.n.], 1994. p. 413–414. Citado 2 vezes nas páginas 39 e 41.
- NIELSEN, J. *Ten usability heuristics*. [S.l.]: <http://www.nngroup.com/articles/ten-usability-heuristics/>(accessed ... , 2005. Citado 3 vezes nas páginas 19, 40 e 41.
- PEREIRA, R. Ihc na especialização em engenharia de software: discussões sobre uma experiência prática. In: *WEIHC*. [S.l.: s.n.], 2013. p. 1–6. Citado na página 17.
- PETRI, G.; WANGENHEIM, C. G. von. A method for the evaluation of the quality of games for computing education. In: *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*. [S.l.: s.n.], 2019. v. 8, n. 1, p. 951. Citado na página 135.
- Roy Daniel. *Vue is an Amazing Framework for 2020. Why?* 2020. <<https://medium.com/weekly-webtips/vue-is-an-amazing-framework-for-2020-why-d8b4e1d01f58>>. Accessed: 2022-03-09. Citado na página 81.
- SCHWABER, K.; BEEDLE, M. *Agile software development with Scrum*. [S.l.]: Prentice Hall Upper Saddle River, 2002. v. 1. Citado na página 35.

- SILVA, M. Proposta de personas para o desenvolvimento de jogos digitais na aprendizagem de ihc. p. 82, 2021. Citado 10 vezes nas páginas 46, 51, 52, 53, 54, 55, 136, 137, 138 e 139.
- SOARES, M. dos S. Metodologias ágeis extreme programming e scrum para o desenvolvimento de software. *Revista Eletrônica de Sistemas de Informação*, v. 3, n. 1, 2004. Citado 2 vezes nas páginas 35 e 36.
- SOMMARIVA, L. et al. Usabilitygame: jogo simulador para apoio ao ensino de usabilidade. In: . [S.l.: s.n.], 2011. Citado 2 vezes nas páginas 18 e 43.
- SOUZA, C. D. *The Semiotic Engineering of Human-computer Interaction*. MIT Press, 2005. (Acting with technology). ISBN 9780262042208. Disponível em: <<https://books.google.com.br/books?id=0yjnotmvtGkC>>. Citado na página 39.
- TAROUCO, L. M. R. et al. Jogos educacionais. *RENOTE: revista novas tecnologias na educação [recurso eletrônico]*. Porto Alegre, RS, 2004. Citado na página 17.
- TOLOMEI, B. V. A gamificação como estratégia de engajamento e motivação na educação. *EaD em Foco*, v. 7, n. 2, set. 2017. Disponível em: <<https://eademfoco.cecierj.edu.br/index.php/Revista/article/view/440>>. Citado na página 17.
- VITALINO, J. F. N.; CASTRO, M. A. N. *Descomplicando o Docker*. [S.l.]: Brasport, 2016. Citado na página 44.
- VR SYS. *As 10 heurísticas de Nielsen - Como elas podem melhorar seu site?* 2019. <<https://www.vrsys.com.br/blog/as-10-heuristicas-de-nielsen-como-elas-podem-melhorar-seu-site>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.
- Wesley Simoes. *As 10 Heurísticas de Nielsen*. 2018. <<https://blog.vulpi.com.br/heuristicas-de-nielsen/>>. Accessed: 2022-03-24. Citado 2 vezes nas páginas 51 e 97.
- ZIMMERMAN, E. Play as research: The iterative design process. *Design research: Methods and perspectives*, MIT Press Cambridge, MA, v. 2003, p. 176–184, 2003. Citado na página 34.

Apêndices

APÊNDICE A – Escolha de tecnologias

Na Tabela 11 está disposto as tecnologias para virtualização do sistema juntamente com suas vantagens e desvantagens.

Tabela 11 – Vantagens e desvantagens das ferramentas de Virtualização do sistema

Tecnologia	Vantagens	Desvantagens
Chef	- Tecnologia completa.	- Configuração complexa; - Difícil para iniciantes.
Puppet	- Tecnologia completa.	- Difícil para iniciantes; - Dependências complexas; - Pago.
Docker	- Fácil de gerenciar; - Fácil de entender; - Tecnologia completa.	- Não é veloz.

Fonte: Elaboração própria (2022)

A escolha pela utilização do Docker se deu por ser uma ferramenta gratuita, de fácil utilização (Burak Karakan, 2020) e por conhecimento/experiência prévia dos autores com a ferramenta.

Na Tabela 12 está disposto as tecnologias para o banco de dados juntamente com suas vantagens e desvantagens.

Tabela 12 – Vantagens e desvantagens para escolha do banco de dados

Tecnologia	Vantagens	Desvantagens
PostgreSQL	- Código aberto; - Facilidade de acesso; - Garantia de estabilidade e segurança.	- Velocidade menor que os demais; - Manutenção.
MySQL	- Código aberto; - Desempenho; - Facilidade de uso.	- Recursos extras são pagos; - Aplicabilidade mais extensa a <i>JavaScript</i> .
SQL Server	- Possui backup de dados; - Amplamente usado no mercado; - Controle de banco de dados centralizados.	- Não é código aberto; - Linguagens para uso limitadas; - Utilizado apenas em <i>Windows</i> .

Fonte: Elaboração própria (2022)

A decisão de utilizar PostgreSQL foi devido a sua aplicabilidade nas tecnologias escolhidas, código livre, curva de aprendizado da equipe (Lucio Chiessi, 2018) e por conhecimento/experiência prévia dos autores com a ferramenta.

Na Tabela 13 está disposto as tecnologias para o *front-end* juntamente com suas vantagens e desvantagens.

Tabela 13 – Vantagens e desvantagens das ferramentas de desenvolvimento do *front-end*

Tecnologia	Vantagens	Desvantagens
Angular	- Robusto; - Documentação detalhada.	- Difícil migração de versões; - Sintaxe mais complexa; - Mutabilidade.
Vue	- Documentação completa; - Facilidade de migração; - Popularidade na comunidade.	- Menos recursos prontos na comunidade; - Dificuldade de aprendizado.
React	- Flexibilidade; - Migração de versões; - Componentes reutilizáveis.	- Documentação reduzida; - Geração de projetos pouco escaláveis.

Fonte: Elaboração própria (2022)

A decisão de utilizar o Vue foi sua facilidade de migração em conjunto a tecnologia de *Back-end*, curva de aprendizado da equipe de desenvolvimento (Roy Daniel, 2020) e por conhecimento/experiência prévia dos autores com a ferramenta.

Na Tabela 14 está disposto as tecnologias para o *back-end* juntamente com suas vantagens e desvantagens.

Tabela 14 – Vantagens e desvantagens das ferramentas de desenvolvimento do *back-end*

Tecnologia	Vantagens	Desvantagens
Flask	- Desempenho mais rápido; - Flexibilidade de aplicação; - Integração com banco de dados.	- Funções assíncronas; - Curva de aprendizado.
Django	- Gestão de banco de dados; - Painel de administração usual; - Arquitetura bem definida.	- Documentação; - Dificuldade de aprendizado.
Express.js	- Grande escala de banco de dados; - Multiplataformas; - Suporte a vários <i>plugins</i> .	- Tratamento de erros.

Fonte: Elaboração própria (2022)

A decisão de utilizar o *Express.js* foi a curva de aprendizado pelo uso de *JavaScript* (Hussain Arif, 2020), sua integração com o banco de dados escolhido e por conhecimento/experiência prévia dos autores com a ferramenta.

Na Tabela 15 está disposto as tecnologias para CI/CD juntamente com suas vantagens e desvantagens.

Tabela 15 – Pontos fortes e fracos e justificativa de escolha da ferramenta de Integração e implantação contínua

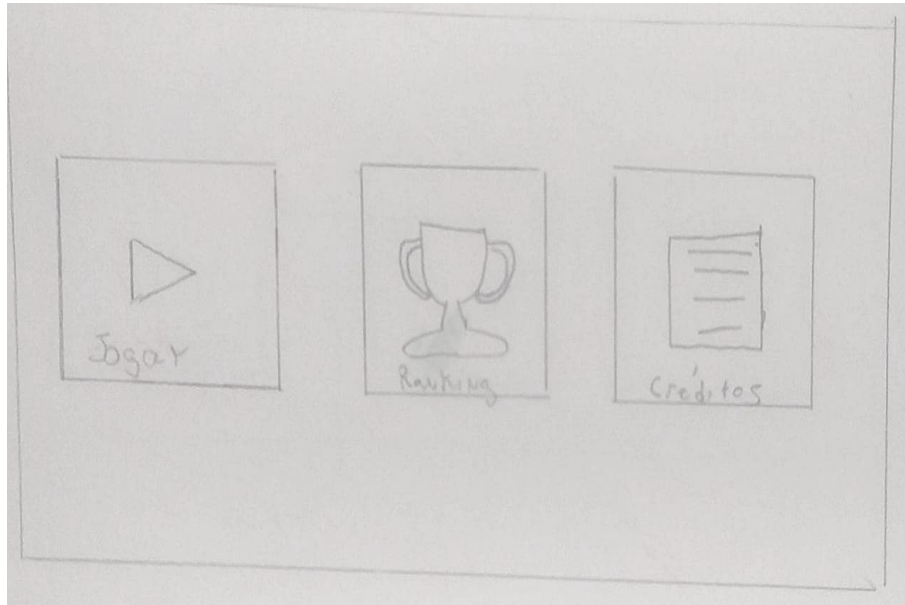
Tecnologia	Vantagens	Desvantagens
Heroku	<ul style="list-style-type: none">- Escalável;- Facilidade no uso	<ul style="list-style-type: none">- Suporte apenas para versão paga- Servidor localizado nos EUA- Tempo de processamento
Vercel	<ul style="list-style-type: none">- Estrutura em JavaScript- Preço- Autonomia	<ul style="list-style-type: none">- Servidor localizado nos EUA
Netlify	<ul style="list-style-type: none">- Facilidade no uso;- Implantação rápida;- Desenvolvimento contínuo	<ul style="list-style-type: none">- Conhecimento prévio da equipe;- Dificuldade em implantações PostgreSQL- Dificuldade em implantações escaláveis

Fonte: Elaboração própria (2022)

A decisão de utilizar as ferramentas de CI/CD Heroku e Vercel foi devido sua autonomia de criar duas implantações que se conectam de forma fácil. Para o *front-end* o Vercel possui maior disponibilidade para linguagem *JavaScript*, facilitando as configurações no código. Para o *back-end* e banco de dados o Heroku permite maior escalabilidade e facilidade no uso, visto que a equipe já havia conhecimento prévio.

APÊNDICE B – Protótipo de baixa fidelidade

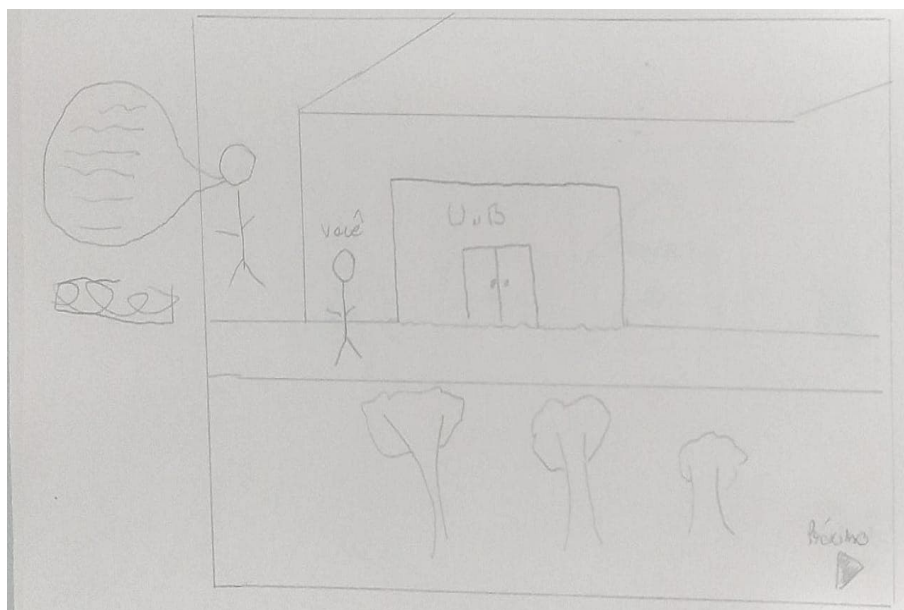
Figura 32 – Tela inicial do jogo



Fonte: Elaboração própria (2022)

Quando clicado em “Jogar” a Figura 32 é aberta, caso o jogador clique em “Ranking” a Figura 33 é aberta, e quando clicado em “Créditos” o jogador não vai para nenhuma Figura, pois esse fluxo não fazia parte do escopo do teste com o protótipo de baixa fidelidade.

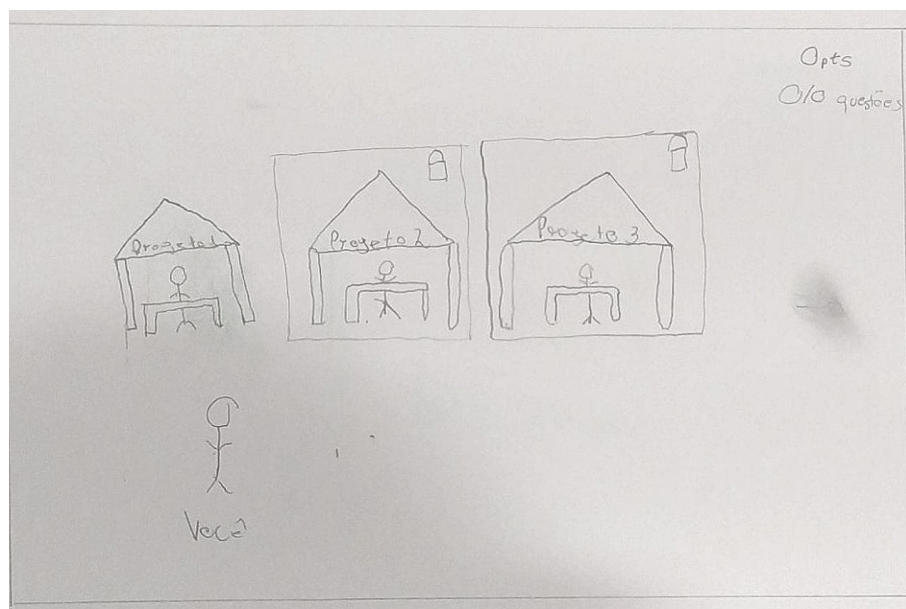
Figura 33 – Tutorial inicial do jogo



Fonte: Elaboração própria (2022)

Quando clicado em “Próximo” o jogador segue para a Figura 34.

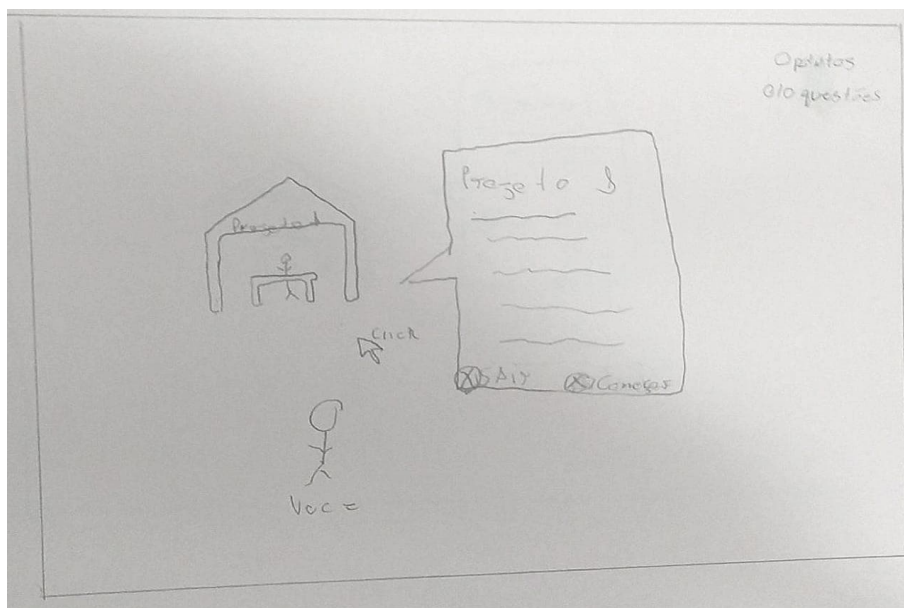
Figura 34 – Escolha dos níveis para se jogar



Fonte: Elaboração própria (2022)

Quando clicado no “Projeto 1” o jogador segue para a Figura 35, tendo os outros dois projetos não habilitados no teste realizado, e quando clicado em voltar o jogador volta para a Figura 32.

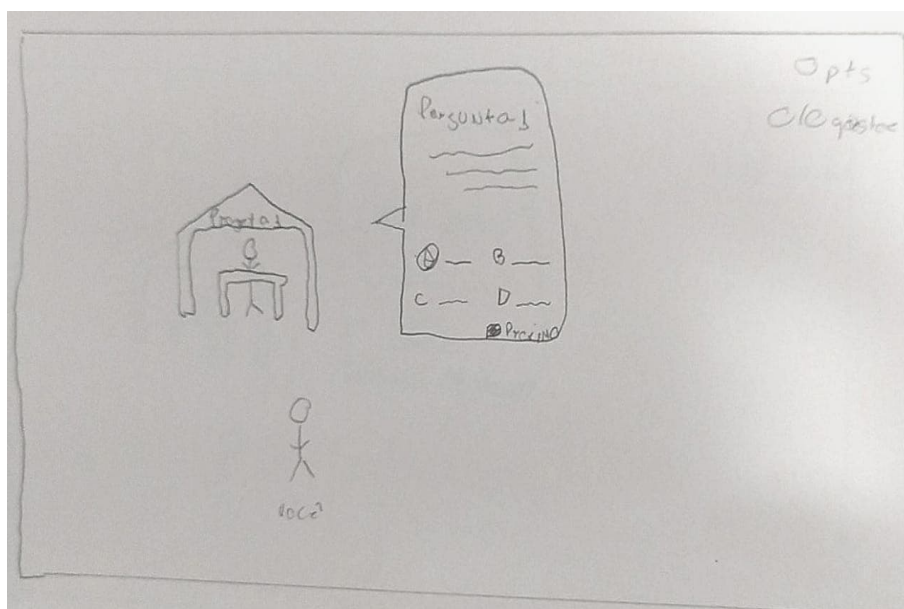
Figura 35 – Início do jogo com explicação do nível um



Fonte: Elaboração própria (2022)

Com o jogo iniciado o jogador pode clicar em “Sair” que voltará para a Figura 32 ou poderá clicar em “Começar” para dar início as perguntas na Figura 36.

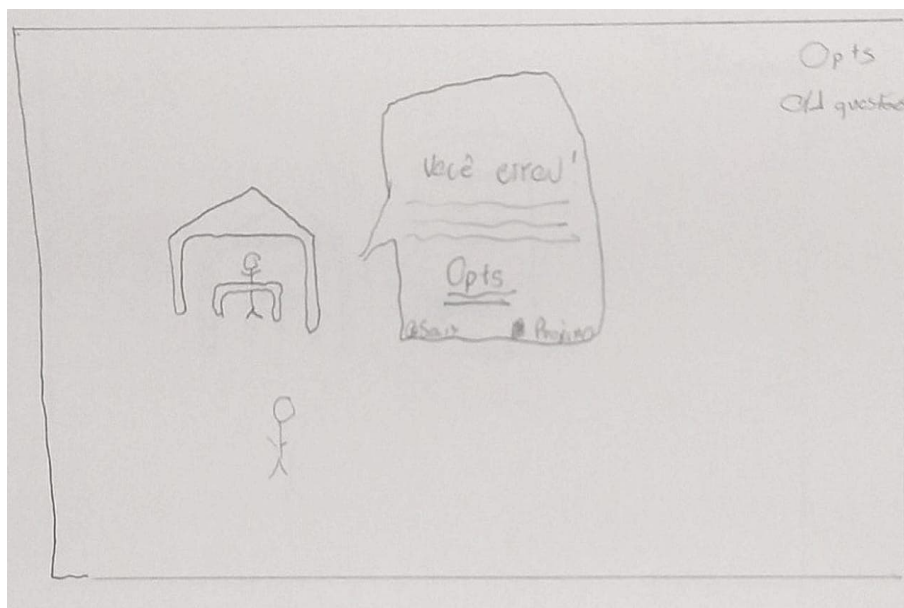
Figura 36 – Início do nível um com a primeira pergunta



Fonte: Elaboração própria (2022)

Com as perguntas iniciadas o jogador clica em “Próximo” para seguir para a Figura 37.

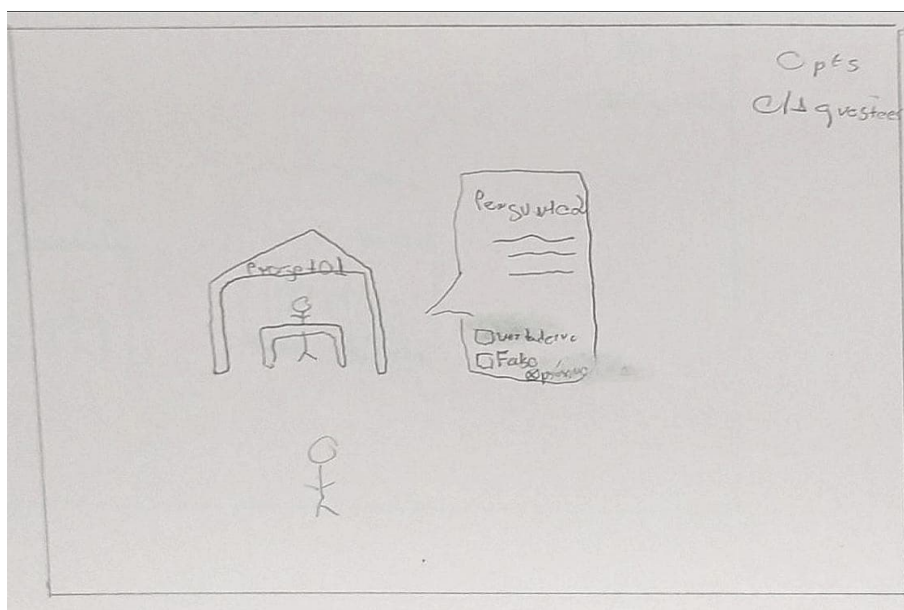
Figura 37 – Feedback quando o usuário errar a pergunta



Fonte: Elaboração própria (2022)

Quando o jogador erra uma pergunta e clica em “Sair” o jogo volta para o menu inicial na Figura 32, se clicado em “Próximo” é dado continuidade no jogo e segue para a Figura 38.

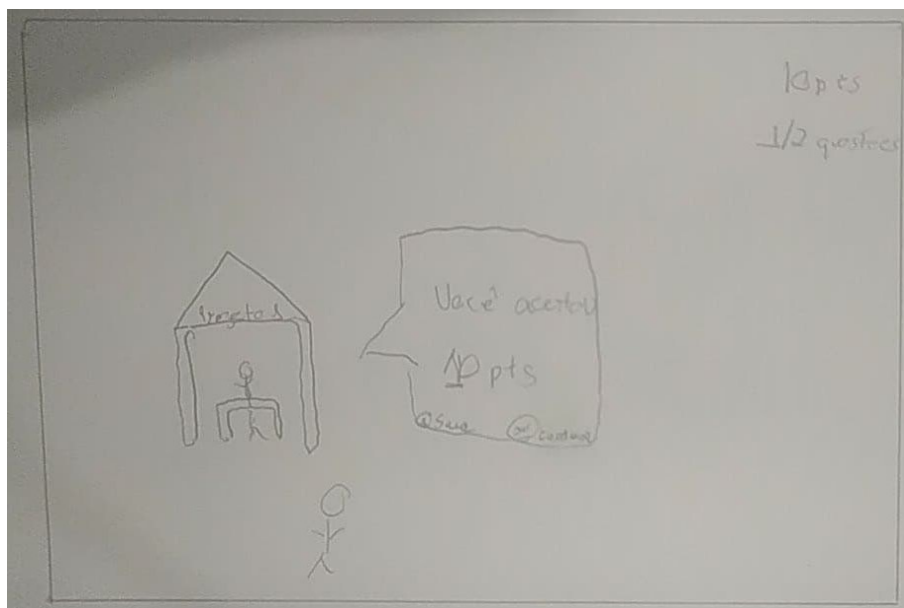
Figura 38 – Segunda pergunta do jogo mostrando que dentro dos níveis terão perguntas binárias e de múltipla escolha



Fonte: Elaboração própria (2022)

Na pergunta seguinte o jogador clica em “Próximo” e segue para a Figura 39.

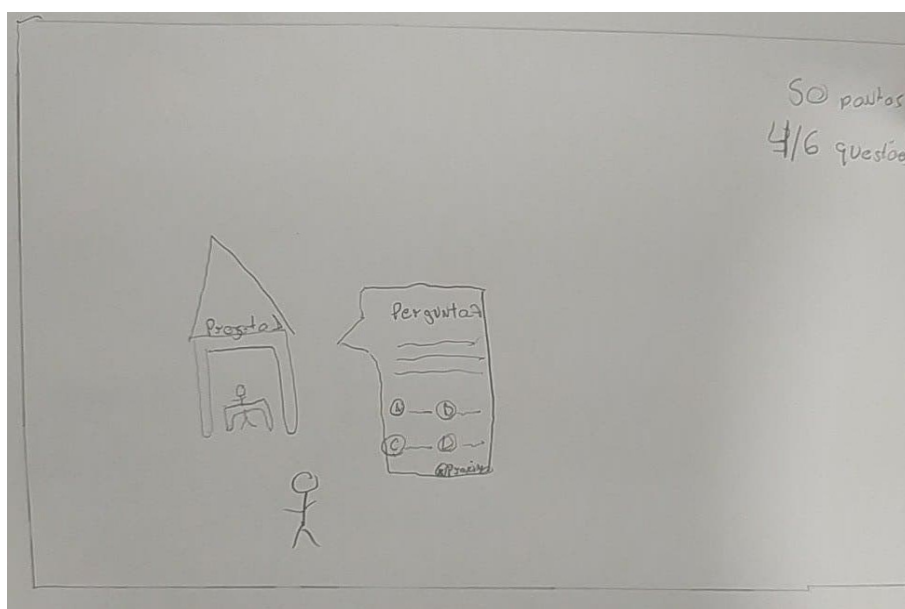
Figura 39 – Mensagem quando o usuário acertar a pergunta



Fonte: Elaboração própria (2022)

Quando o jogador acerta uma pergunta e clica em “Sair” o jogo volta para o menu inicial na Figura 32, se clicado em “Próximo” é dado continuidade no jogo e segue para a Figura 40.

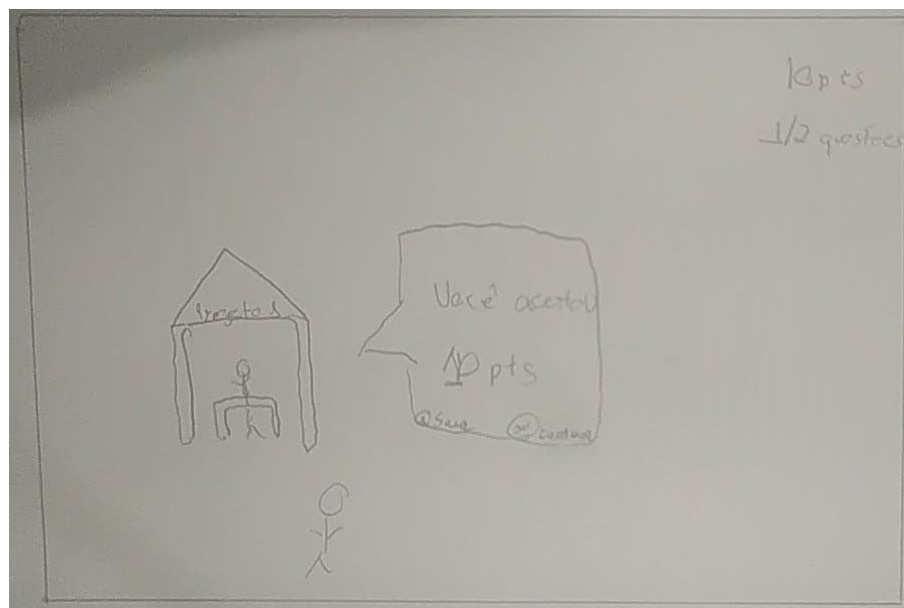
Figura 40 – Sequência de perguntas do jogo



Fonte: Elaboração própria (2022)

Na pergunta seguinte o jogador clica em “Próximo” e segue para a Figura 41.

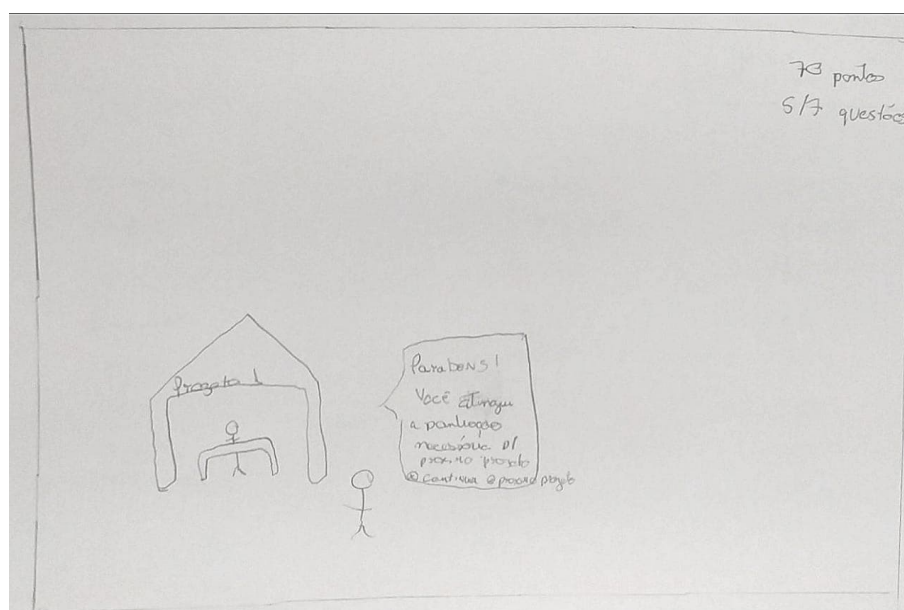
Figura 41 – Mensagem de acerto do usuário



Fonte: Elaboração própria (2022)

Quando o jogador acerta uma pergunta e clica em “Sair” o jogo volta para o menu inicial na Figura 32, se clicado em “Próximo” é dado continuidade no jogo e segue para a Figura 42.

Figura 42 – Feedback ao final do nível 1 dizendo se o usuário atingiu o necessário para passar de nível ou não



Fonte: Elaboração própria (2022)

Quando o jogador finaliza o projeto ele pode clicar em “Continuar” para ir à Figura 34 e a opção de “Continuar jogando” não está habilitada nesse protótipo por não fazer

parte do escopo do teste realizado.

Figura 43 – Ranking do jogo

Nome	↑ Pontos	questões %
Pedro	2500 pts	780%
José	2200	85%
Emmanuel	2150	70%
Sulra	1800	65%
Léo	1350	80%
André	820	50%
Fabiana	780	65%
:		

10 pts
1/2 questões

← Voltar

Fonte: Elaboração própria (2022)

Quando clicado em “Voltar” o jogador irá para o menu principal na Figura 32.

APÊNDICE C – Protótipo de alta fidelidade

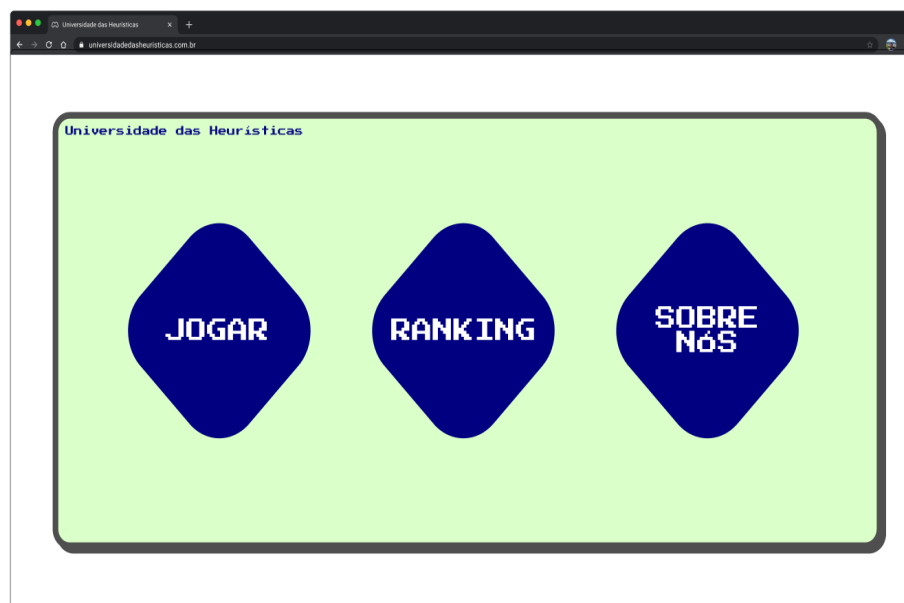
Figura 44 – Página inicial do jogo



Fonte: Elaboração própria (2022)

Quando clicado em “Dê o play” a Figura 45 é aberta.

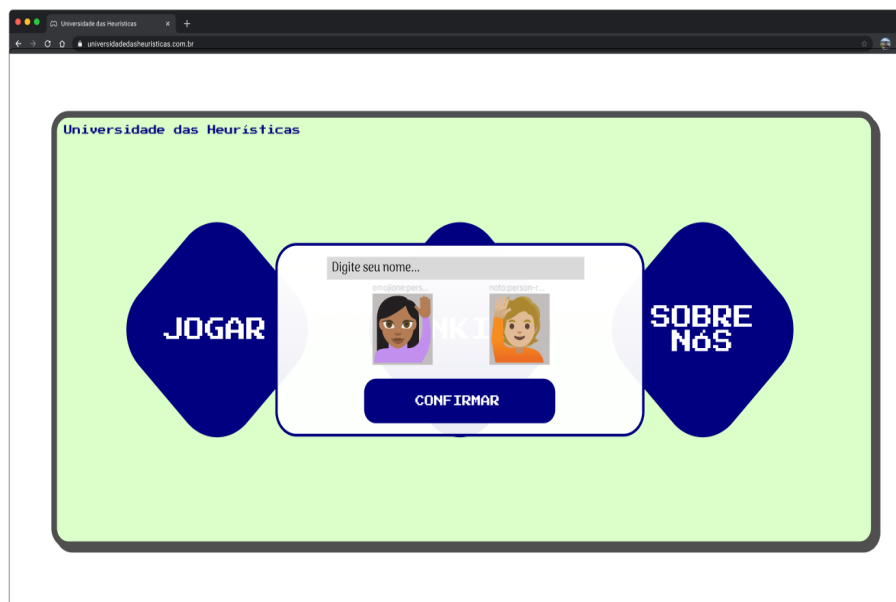
Figura 45 – Menu inicial para acessar ranking e jogar



Fonte: Elaboração própria (2022)

Quando clicado em “Jogar” a Figura 46 é aberta, caso o jogador clique em “Ranking” a Figura 53 é aberta, e quando clicado em “Créditos” o jogador não vai para nenhuma figura, pois esse fluxo não fazia parte do escopo do teste com o protótipo de alta fidelidade.

Figura 46 – Caixa de diálogo que é mostrado após clicar em jogar



Fonte: Elaboração própria (2022)

Quando digitado um nome e clicado em “Confirmar” o jogador segue para a Figura 47.

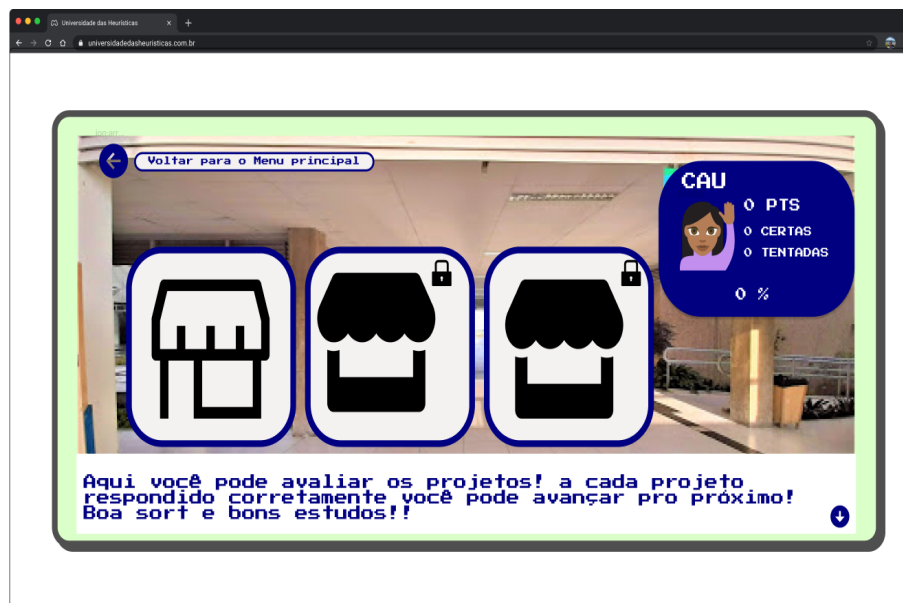
Figura 47 – Introdução ao jogo e uma breve história



Fonte: Elaboração própria (2022)

Quando clicado na seta do canto inferior direito do tutorial, o jogador vê o restante do tutorial e posteriormente segue para a Figura 48.

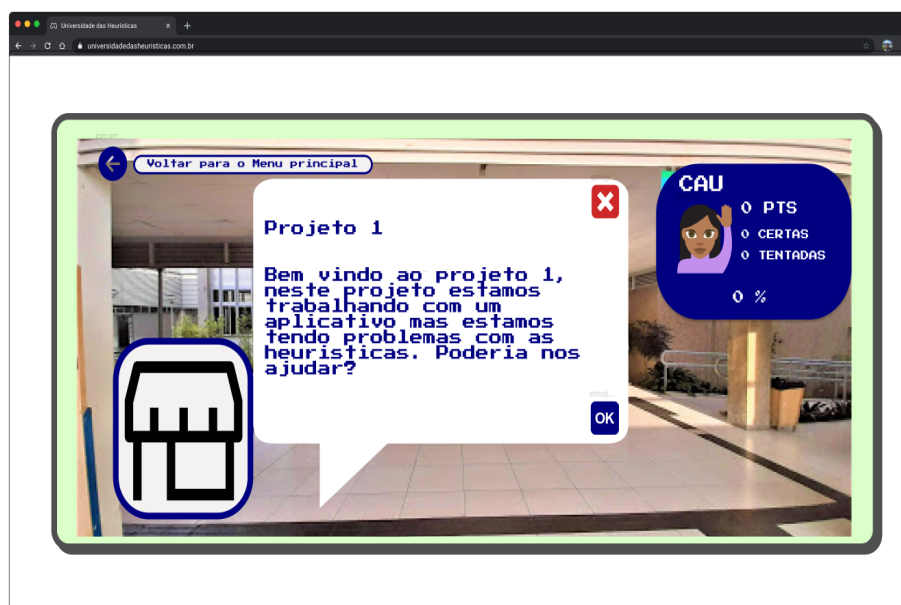
Figura 48 – Apresentação da tela do jogo com a disposição dos projetos



Fonte: Elaboração própria (2022)

Com o jogo iniciado o jogador pode clicar na seta do canto superior esquerdo que voltará para a Figura 45 ou poderá clicar na seta do canto inferior direito para dar início as perguntas na Figura 49.

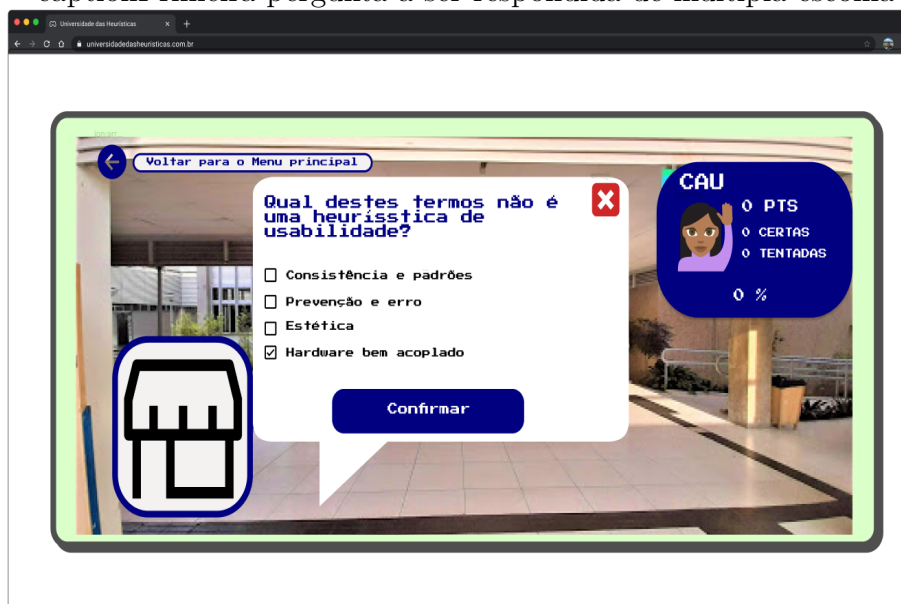
Figura 49 – Breve introdução do primeiro projeto



Fonte: Elaboração própria (2022)

Quando clicado em “OK” o jogador segue para a Figura ??, caso seja clicado na seta do canto superior esquerdo, voltará para a Figura 45.

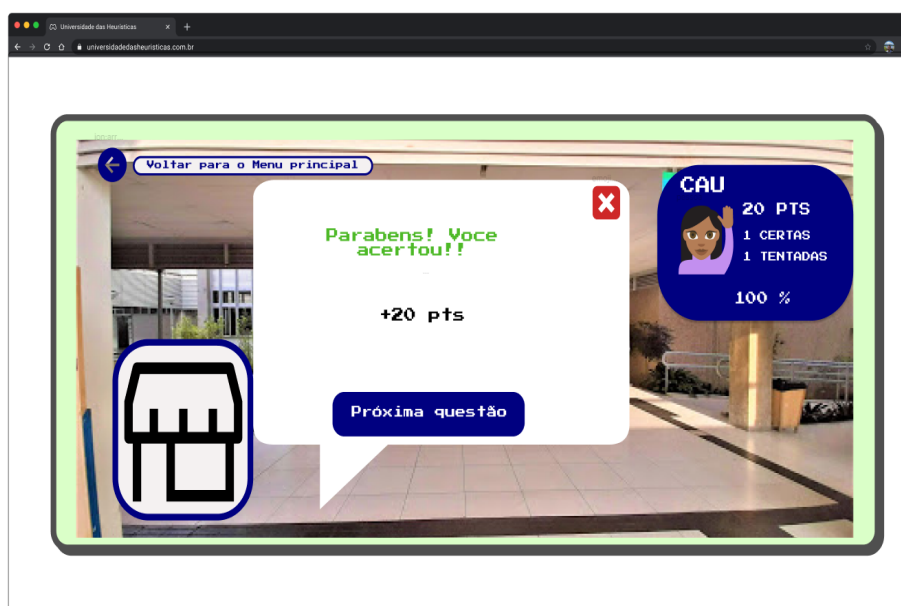
captionPrimeira pergunta a ser respondida de múltipla escolha



Fonte: Elaboração própria (2022)

Com as perguntas iniciadas o jogador clica em “Confirmar” para seguir para a Figura 50, caso seja clicado na seta do canto superior esquerdo, voltará para a Figura 45.

Figura 50 – Feedback de pontuação quando acerta a resposta

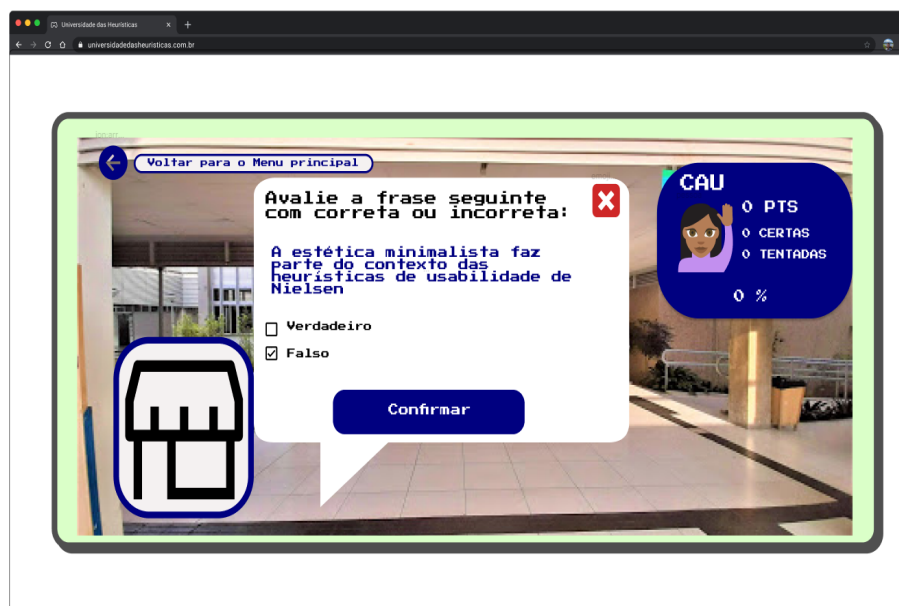


Fonte: Elaboração própria (2022)

Quando o jogador acerta uma pergunta e clica em “Próxima questão” é dado continuidade no jogo e segue para a Figura 51, caso seja clicado na seta do canto superior

esquerdo, voltará para a Figura 45.

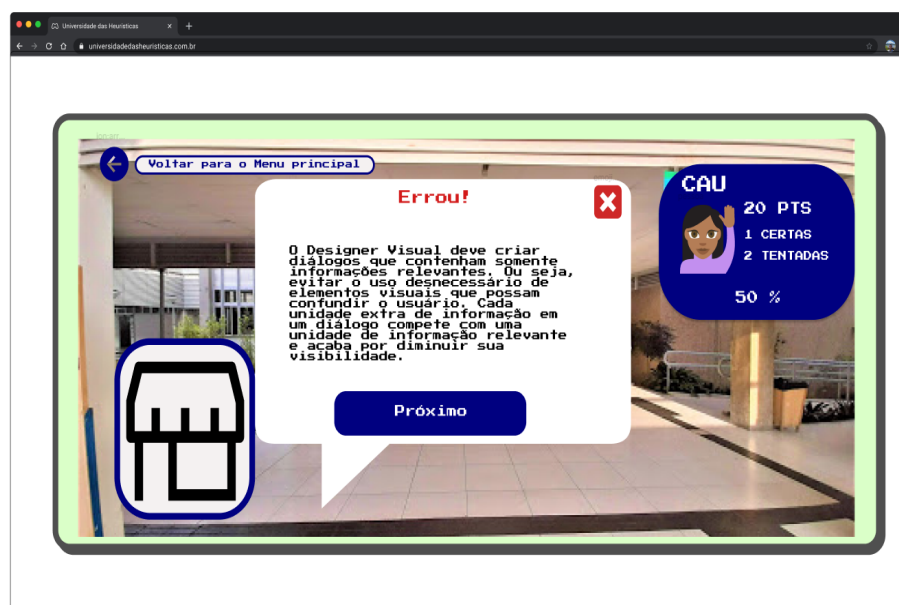
Figura 51 – Segunda pergunta a ser respondida de verdadeiro ou falso



Fonte: Elaboração própria (2022)

Na pergunta seguinte o jogador clica em “Confirmar” e segue para a Figura 41, caso seja clicado na seta do canto superior esquerdo, voltará para a Figura 45.

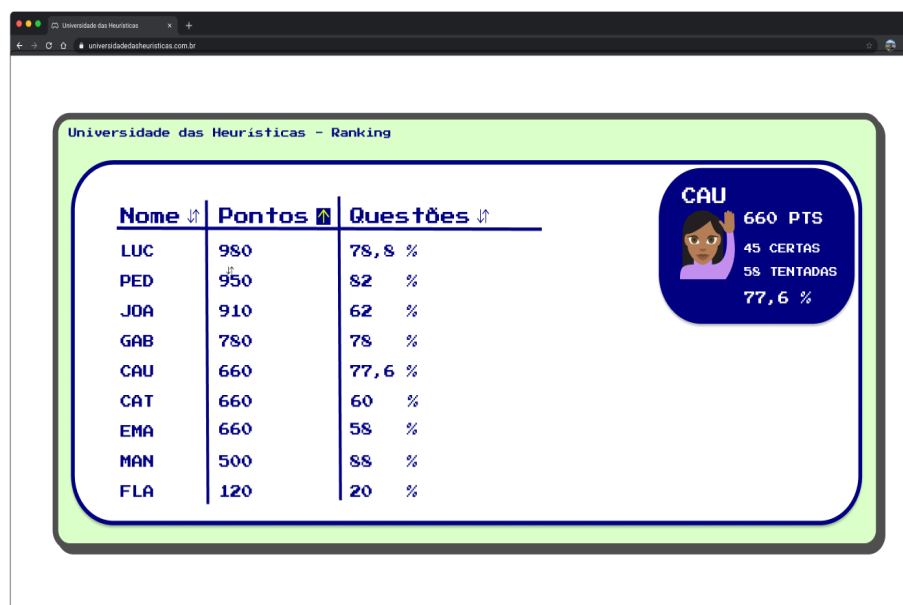
Figura 52 – Feedback quando erra a questão



Fonte: Elaboração própria (2022)

Quando o jogador erra uma pergunta e clica em na seta do canto superior esquerdo, voltará para a Figura 45, se clicado em “Próximo” é dado continuidade no jogo.

Figura 53 – Página de ranking acessada ao menu principal



Fonte: Elaboração própria (2022)

Quando clicado em “Voltar” o jogador irá para o menu principal na Figura 45.

APÊNDICE D – Banco de dados das perguntas

O jogo Universidade das Heurísticas possui um total 60 perguntas no banco de dados, divididas igualmente entre tipos, verdadeiro ou falso e múltipla escolha (30 perguntas de verdadeiro ou falso e 30 perguntas de múltipla escolha). Foram subdivididas entre os níveis fácil, médio e difícil, divididas igualmente entre os níveis com 20 perguntas em cada. Na Tabela 16 pode-se ver a listagem das figuras com as perguntas, qual a dificuldade delas e qual a referência utilizada para criação.

Tabela 16 – Tabela com perguntas, níveis e referências do banco de dados

Perguntas	Figura referência das perguntas	Nível	Referência
1	54	Fácil	(VR SYS, 2019)
2	55	Fácil	(VR SYS, 2019)
3	56	Fácil	(Gabrielle Pinhata, 2021)
4	57	Fácil	(Gabrielle Pinhata, 2021)
5	58	Fácil	(Marco Bruno, 2019)
6	59	Fácil	(Marco Bruno, 2019)
7	60	Fácil	(Gabriel Moma, 2017)
8	61	Fácil	(Gabriel Moma, 2017)
9	62	Fácil	(Editorial Aela, 2019)
10	63	Fácil	(Editorial Aela, 2019)
11, 12, 13 e 14	64	Fácil	(Henrique Carvalho, 2020)
15, 16, 17 e 18	65	Fácil	(Henrique Carvalho, 2020)
19 e 20	66	Fácil	(Henrique Carvalho, 2020)
21	67	Médio	(Gabrielle Pinhata, 2021)
22	68	Médio	(Gabrielle Pinhata, 2021)
23	69	Médio	(Gabrielle Pinhata, 2021)
24	70	Médio	(Gabrielle Pinhata, 2021)
25	71	Médio	(Gabrielle Pinhata, 2021)
26	72	Médio	(Editorial Aela, 2019)
27	73	Médio	(Editorial Aela, 2019)
28	74	Médio	(Editorial Aela, 2019)
29	75	Médio	(Wesley Simoes, 2018)
30	76	Médio	(Gabriel Moma, 2017)
31, 32, 33 e 34	77	Médio	(Henrique Carvalho, 2020)
35, 36, 37 e 38	78	Médio	(Henrique Carvalho, 2020)
39 e 40	79	Médio	(Henrique Carvalho, 2020)
41	80	Difícil	(Gabriel Moma, 2017)
42	81	Difícil	(Gabriel Moma, 2017)
43	82	Difícil	(Gabriel Moma, 2017)
44	83	Difícil	(Marco Bruno, 2019)
45	84	Difícil	(Marco Bruno, 2019)
46	85	Difícil	(Marco Bruno, 2019)
47	86	Difícil	(Marco Bruno, 2019)
48	87	Difícil	(Marco Bruno, 2019)
49	88	Difícil	(Gabriel Moma, 2017)
50	89	Difícil	(Marco Bruno, 2019)
51, 52, 53 e 54	90	Difícil	Elaboração própria (2022)
55, 56, 57 e 58	91	Difícil	Elaboração própria (2022)
59 e 60	92	Difícil	Elaboração própria (2022)

Fonte: Elaboração própria (2022)

Na Tabela 16 estão listadas as perguntas Na Seção D.1 estão todas as perguntas do nível um separadas nas Subseções D.1.1 e D.1.2 que possuem perguntas do tipo múltipla

escolha e verdadeiro ou falso, respectivamente. Na Seção D.2 estão todas as perguntas do nível dois separadas nas Subseções D.2.1 e D.2.2 que possuem perguntas do tipo múltipla escolha e verdadeiro ou falso, respectivamente. E na Seção D.3 estão todas as perguntas do nível três separadas nas Subseções D.3.1 e D.3.2 que possuem perguntas do tipo múltipla escolha e verdadeiro ou falso, respectivamente.

As perguntas apresentadas nas figuras possuem o nível de dificuldade, enunciado, resposta marcada em verde, *feedback* para resposta errada e no caso das perguntas do tipo múltipla escolha possui as opções. As perguntas estão separadas entre nível e tipo da pergunta, que podem ser conferidas nas seções seguintes.

D.1 Perguntas do nível um

D.1.1 Perguntas de múltipla escolha

Figura 54 – Pergunta do nível um de múltipla escolha



(FÁCIL) Qual heurística melhor representa a imagem abaixo:

Home | Seu carrinho **Breadcrumb**

Resumo do carrinho
Seu carrinho de compras contém: 1 produto

Etapas do checkout

01. Resumo 02. Entrar 03. Endereço 04. Frete 05. Pagamento

Produto	Descrição	Disponibilidade	Preço Unitário	Qty	Total
	Camiseta Básica Infantil Branca Em Malha Algodão SKU:SS_ALG_INF_BR_2 Tamanho:2, Cores:Branco	Pronta entrega	R\$ 29,80	- 1 + 	R\$ 29,80
Cupons de desconto				Total de produtos	R\$ 29,80


a) Visibilidade do status do sistema
b) Compatibilidade entre o sistema e o mundo real
c) Controle e liberdade para o usuário
d) Consistência e padronização

FEEDBACK: A heurística que melhor representa a imagem é a de visibilidade do status do sistema. A imagem mostra as etapas do checkout em um processo na compra de um produto em uma loja virtual e essas etapas representam o status do processo de compra.

Fonte: Elaboração própria (2022)

Figura 55 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:



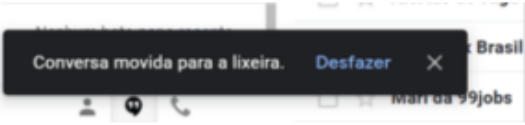
a) Compatibilidade entre o sistema e o mundo real
 b) Ajuda e documentação
 c) Visibilidade do status do sistema
 d) **Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros**

FEEDBACK: A heurística que melhor representa a imagem é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros. A imagem mostra a tela de recuperação do sistema operacional windows, onde ocorreu um problema, mas o usuário pode resolvê-lo reiniciando o computador ou verificando outras opções avançadas de resolução.

Fonte: Elaboração própria (2022)

Figura 56 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:





a) Reconhecimento em vez de memorização
 b) Eficiência e flexibilidade de uso
 c) Estética e design minimalista
 d) **Controle e liberdade para o usuário**

FEEDBACK: A heurística que melhor representa a imagem é a de controle e liberdade para o usuário. A imagem mostrada permite que o usuário desfça de uma ação, caso ela tenha sido realizada intencionalmente ou em caso de arrependimento, isso dá controle sobre as ações do usuário no sistema.

Fonte: Elaboração própria (2022)

Figura 57 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:

Pressione esta tecla	Para fazer isto
Ctrl + X	Recortar o item selecionado.
Ctrl + C (ou Ctrl + Insert)	Copiar o item selecionado.
Ctrl + V (ou Shift + Insert)	Colar o item selecionado.
Ctrl + Z	Desfazer uma ação.
Alt + Tab	Alternar aplicativos abertos.
Alt + F4	Fechar o item ativo ou sair do aplicativo ativo.
Tecla do logotipo do Windows  + L	Bloquear seu computador.
Tecla do logotipo do Windows  + D	Exibir e ocultar a área de trabalho.

a) **Visibilidade do status do sistema**

b) Ajuda e documentação

c) **Eficiência e flexibilidade de uso**


d) Compatibilidade entre o sistema e o mundo real

FEEDBACK: A heurística que melhor representa a imagem é a de eficiência e flexibilidade de uso. A imagem mostra alguns atalhos que podem ser utilizados em um sistema para diminuir o tempo gasto do usuário para realizar algumas ações, tornando o sistema eficiente para usuários mais avançados.

Fonte: Elaboração própria (2022)

Figura 58 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:




a) Eficiência e flexibilidade de uso
b) **Controle e liberdade para o usuário**
c) Estética e design minimalista
d) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

FEEDBACK: A heurística que melhor representa a imagem é a de controle e liberdade para o usuário. A imagem mostrada permite que o usuário curta, compartilhe ou encaminhe a postagem da imagem, sendo que caso ele deseje discutir ou remover o compartilhamento a plataforma permite que o usuário o faça.

Fonte: Elaboração própria (2022)

Figura 59 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:



Aproximadamente 635.000.000 resultados (0,27 segundos)

Exibindo resultados para **user experience**
Em vez disso, pesquisar por **user experience**

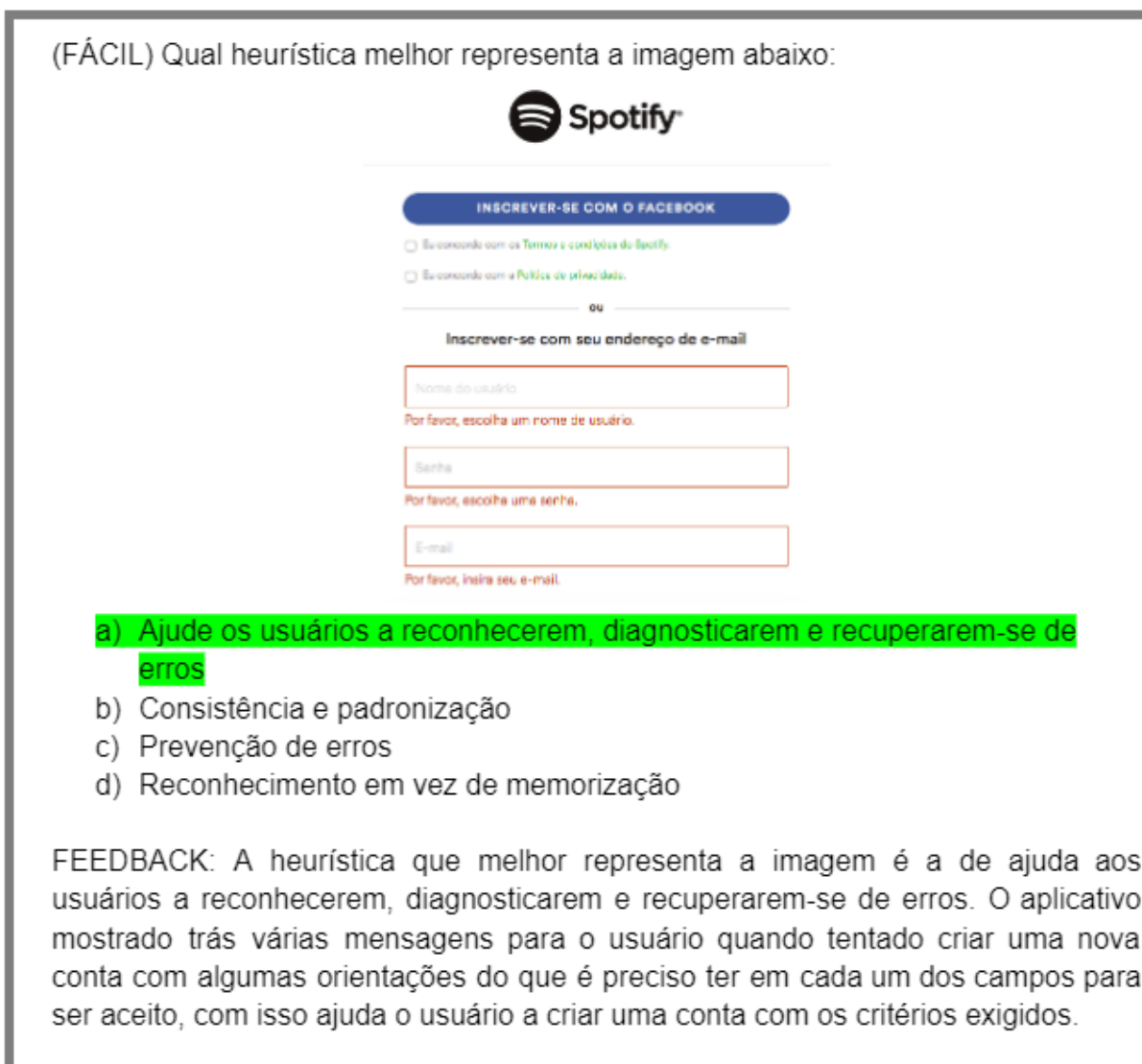
- a) Prevenção de erros
- b) Ajuda e documentação
- c) Visibilidade do status do sistema
- d) Compatibilidade entre o sistema e o mundo real

FEEDBACK: A heurística que melhor representa a imagem é a de prevenção de erros. A imagem mostra o sistema de busca fazer uma sugestão de correção na busca realizada pelo usuário escrita de forma errada.

Fonte: Elaboração própria (2022)

Figura 60 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:



The image shows the Spotify sign-up page. At the top, there is the Spotify logo. Below it, there is a blue button that says "INSCREVER-SE COM O FACEBOOK". Underneath the button, there are two radio buttons: "Se inscreva com os Termos e condições do Spotify" and "Se inscreva com a Política de privacidade". Below these, there is a horizontal line with the word "ou" in the center. Underneath the line, there is a heading "Inscrever-se com seu endereço de e-mail". Below this heading, there are three input fields: "Nome do usuário", "Senha", and "E-mail". Each input field has a red border and a red error message below it: "Por favor, escolha um nome de usuário.", "Por favor, escolha uma senha.", and "Por favor, insira seu e-mail." respectively.

a) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

b) Consistência e padronização

c) Prevenção de erros

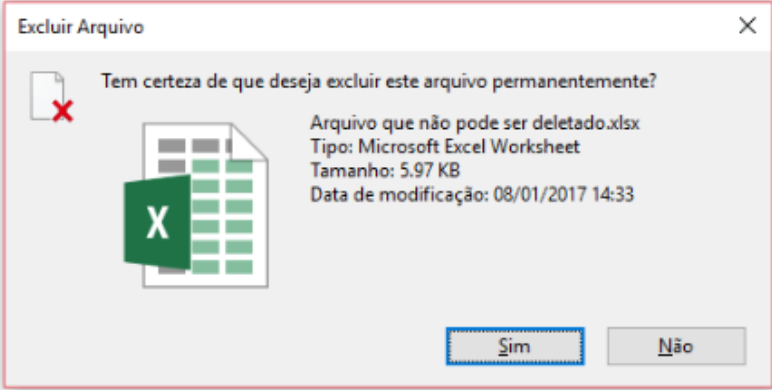
d) Reconhecimento em vez de memorização

FEEDBACK: A heurística que melhor representa a imagem é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros. O aplicativo mostrado trás várias mensagens para o usuário quando tentado criar uma nova conta com algumas orientações do que é preciso ter em cada um dos campos para ser aceito, com isso ajuda o usuário a criar uma conta com os critérios exigidos.

Fonte: Elaboração própria (2022)

Figura 61 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:



a) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

b) Ajuda e documentação

c) Visibilidade do status do sistema


d) **Prevenção de erros**

FEEDBACK: A heurística que melhor representa a imagem é a de prevenção de erros. A imagem mostra uma mensagem se o usuário deseja realmente excluir um arquivo após ter clicado para excluir o arquivo, com isso impede o usuário de fazer algo caso se arrependa ou tenha clicado sem intenção.

Fonte: Elaboração própria (2022)

Figura 62 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:



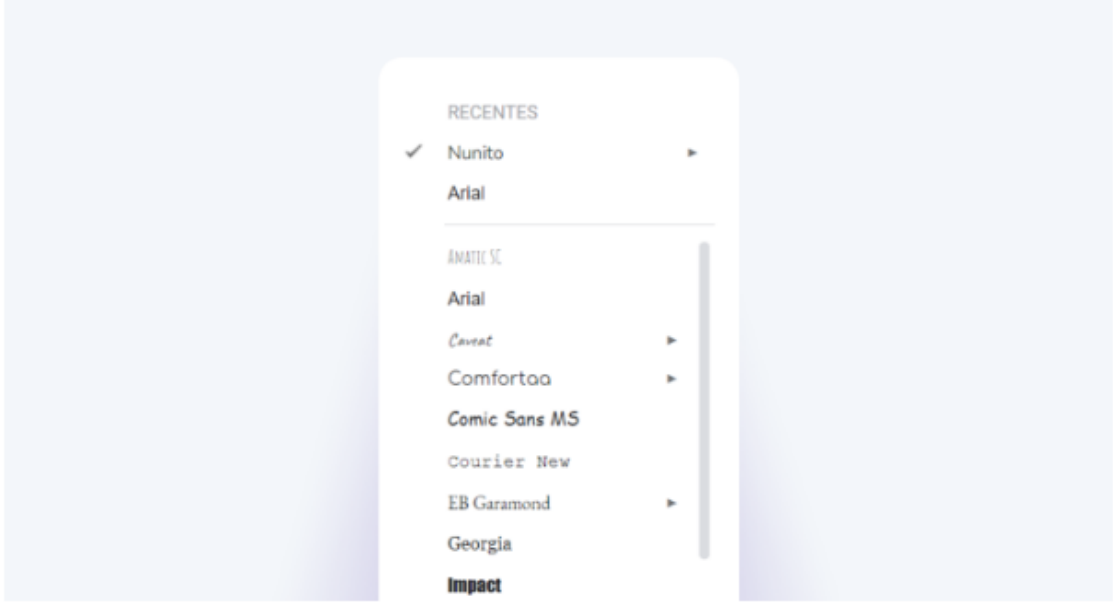
a) Visibilidade do status do sistema
b) Compatibilidade entre o sistema e o mundo real
c) Controle e liberdade para o usuário
d) Consistência e padronização

FEEDBACK: A heurística que melhor representa a imagem é a de compatibilidade entre o sistema e o mundo real. A imagem mostrada trás elementos que na vida real representam exatamente a funcionalidade de cada um dos botões, ou seja, os botões falam a mesma língua do mundo real.

Fonte: Elaboração própria (2022)

Figura 63 – Pergunta do nível um de múltipla escolha

(FÁCIL) Qual heurística melhor representa a imagem abaixo:



a) Reconhecimento em vez de memorização

b) Eficiência e flexibilidade de uso

c) Estética e design minimalista

d) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

FEEDBACK: A heurística que melhor representa a imagem é a de reconhecimento em vez de memorização. O sistema reconhece as fontes mais recentes utilizadas pelo usuário e as destaca para que possa ver as últimas usadas.

Fonte: Elaboração própria (2022)

D.1.2 Perguntas de verdadeiro ou falso

Figura 64 – Perguntas de nível um de verdadeiro ou falso

<p>(FÁCIL) A heurística de visibilidade do status do sistema se refere à importância dos feedbacks instantâneos aos usuários, mantendo-os informados sobre o que está acontecendo no momento da interação, qual o status do momento e como ele deve conduzir seus próximos passos no processo de sua experiência. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de visibilidade do status do sistema, ou seja, a afirmação estava correta.</p>
<p>(FÁCIL) A heurística de compatibilidade entre o sistema e o mundo real se refere à necessidade do sistema ter a mesma linguagem (palavras, frases, imagens e conceitos) que o usuário utiliza no dia a dia. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de compatibilidade entre o sistema e o mundo real, ou seja, a afirmação estava correta.</p>
<p>(FÁCIL) A heurística de controle e liberdade para o usuário se refere à liberdade para realizar ações que o usuário deseja dentro de um sistema. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de controle e liberdade para o usuário, ou seja, a afirmação estava correta.</p>
<p>(FÁCIL) A heurística de consistência e padronização se refere a um padrão que a interface e linguagem de um sistema deve possuir para não confundir os usuários. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de consistência e padronização, ou seja, a afirmação estava correta.</p>

Fonte: Elaboração própria (2022)

Figura 65 – Perguntas do nível um de verdadeiro ou falso

<p>(FÁCIL) A heurística de prevenção de erros se refere à prevenção de qualquer ação descuidada que um usuário possa cometer durante o uso de um sistema. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de prevenção de erros, ou seja, a afirmação estava correta.</p>
<p>(FÁCIL) A heurística de reconhecimento em vez de memorização se refere ao sistema disponibilizar lembretes com informações úteis sobre as funcionalidades de um sistema. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de reconhecimento em vez de memorização, ou seja, a afirmação estava correta.</p>
<p>(FÁCIL) A heurística de eficiência e flexibilidade de uso se refere à flexibilidade de dispor informações detalhadas para os usuários poderem customizar suas ações e criar atalhos e também deve ser um sistema eficaz permitindo personalizar ações frequentes e tendo preenchimentos automáticos. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de eficiência e flexibilidade de uso, ou seja, a afirmação estava correta.</p>
<p>(FÁCIL) A heurística de estética e design minimalista se refere ao sistema possuir elementos visuais diretos e simples, sem o uso em excesso ou desnecessário de elementos. Essa sentença é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de estética e design minimalista, ou seja, a afirmação estava correta.</p>

Fonte: Elaboração própria (2022)

Figura 66 – Perguntas do nível um de verdadeiro ou falso

(FÁCIL) A heurística de ajuda os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros se refere ao sistema em casos de erros durante o uso, ajudarem os usuários a entender o problema ocorrido, além de dispor informações para poder solucionar o problema. Essa sentença é verdadeira ou falsa? **Verdadeiro**
FEEDBACK: A heurística da afirmação é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros, ou seja, a afirmação estava correta.

(FÁCIL) A heurística de ajuda e documentação se refere aos sistemas possuírem uma seção de ajuda para os usuários poderem obter respostas rápidas. Essa sentença é verdadeira ou falsa? **Verdadeiro**
FEEDBACK: A heurística da afirmação é a de ajuda e documentação, ou seja, a afirmação estava correta.

Fonte: Elaboração própria (2022)

D.2 Perguntas do nível dois

D.2.1 Perguntas de múltipla escolha

Figura 67 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



a) **Visibilidade do status do sistema**

b) Consistência e padronização

c) Prevenção de erros

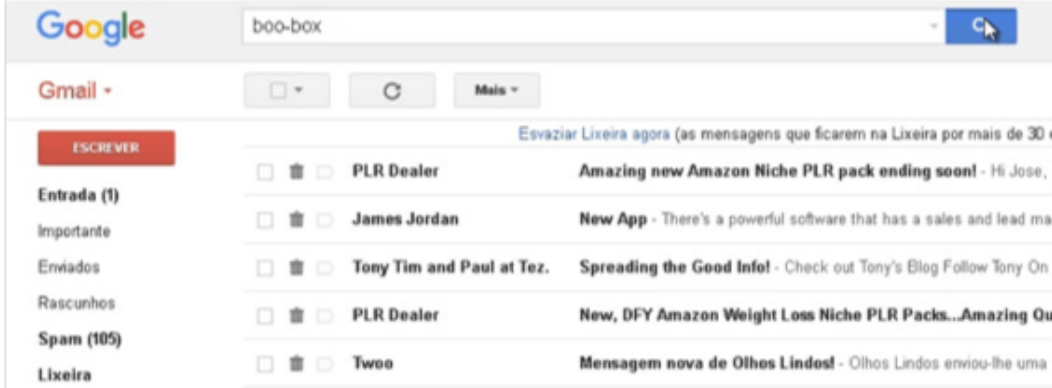
d) Reconhecimento em vez de memorização

FEEDBACK: A heurística que melhor representa a imagem é a de visibilidade do status do sistema. A imagem mostra a barra lateral de uma plataforma de vídeo que mostra os vídeos já assistidos, o vídeo que está sendo assistido e o vídeo que será mostrado na sequência, ou seja, mostra o status da playlist de vídeos.

Fonte: Elaboração própria (2022)

Figura 68 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:




a) Prevenção de erros
b) Reconhecimento em vez de memorização
c) Eficiência e flexibilidade de uso
d) **Compatibilidade entre o sistema e o mundo real**

FEEDBACK: A heurística que melhor representa a imagem é a de compatibilidade entre o sistema e o mundo real. O serviço de correio eletrônico mostrado faz o uso de elementos como a lupa e a lixeira que tem funções no sistema exatamente dos objetos físicos. Há aqui uma compatibilidade entre as funcionalidades dos elementos com a versão física deles.

Fonte: Elaboração própria (2022)

Figura 69 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



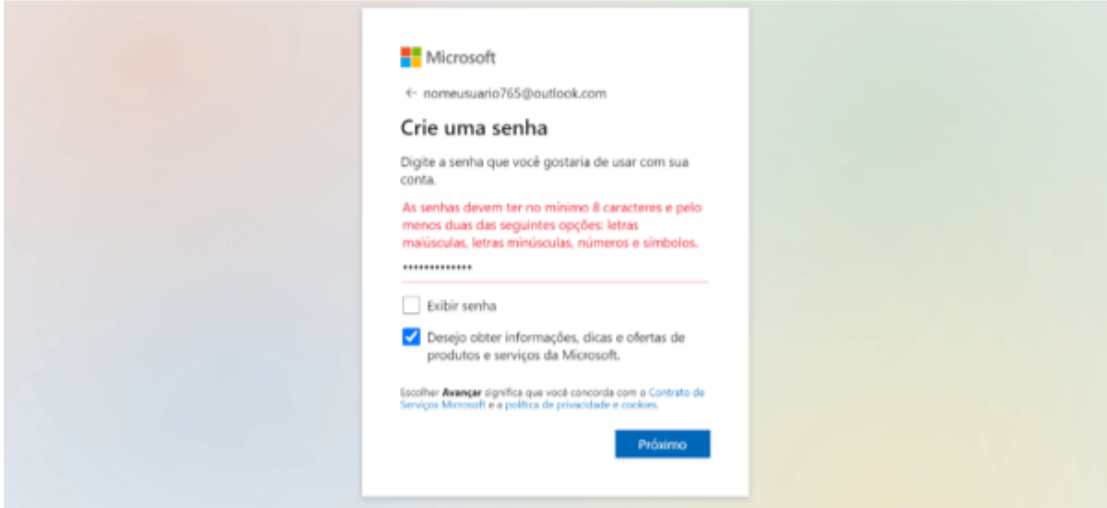
a) Compatibilidade entre o sistema e o mundo real
b) Visibilidade do status do sistema
c) **Estética e design minimalista**
d) Controle e liberdade para o usuário

FEEDBACK: A heurística que melhor representa a imagem é a de estética e design minimalista. O site mostrado apresenta uma estética e design bem simples e objetivo e consegue trazer todas as informações que se quer mostrar para os usuários.

Fonte: Elaboração própria (2022)

Figura 70 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



a) Compatibilidade entre o sistema e o mundo real

b) **Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros**

c) Controle e liberdade para o usuário

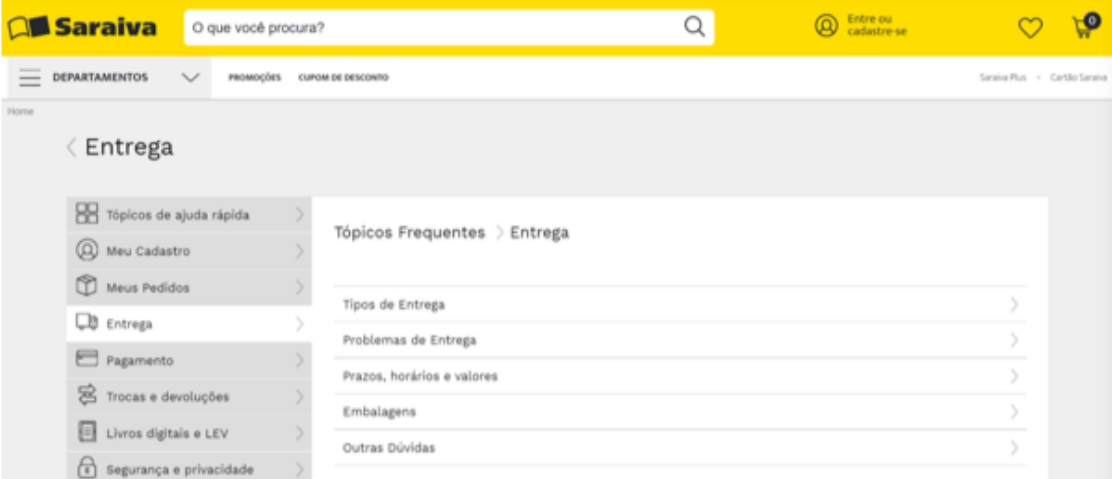
d) Consistência e padronização

FEEDBACK: A heurística que melhor representa a imagem é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros. A imagem mostrada trás uma mensagem para o usuário quando tentado criar uma senha para seu e-mail com algumas orientações do que é preciso ter em uma senha para ser aceito, com isso ajuda o usuário a definir uma senha com os critérios exigidos.

Fonte: Elaboração própria (2022)

Figura 71 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



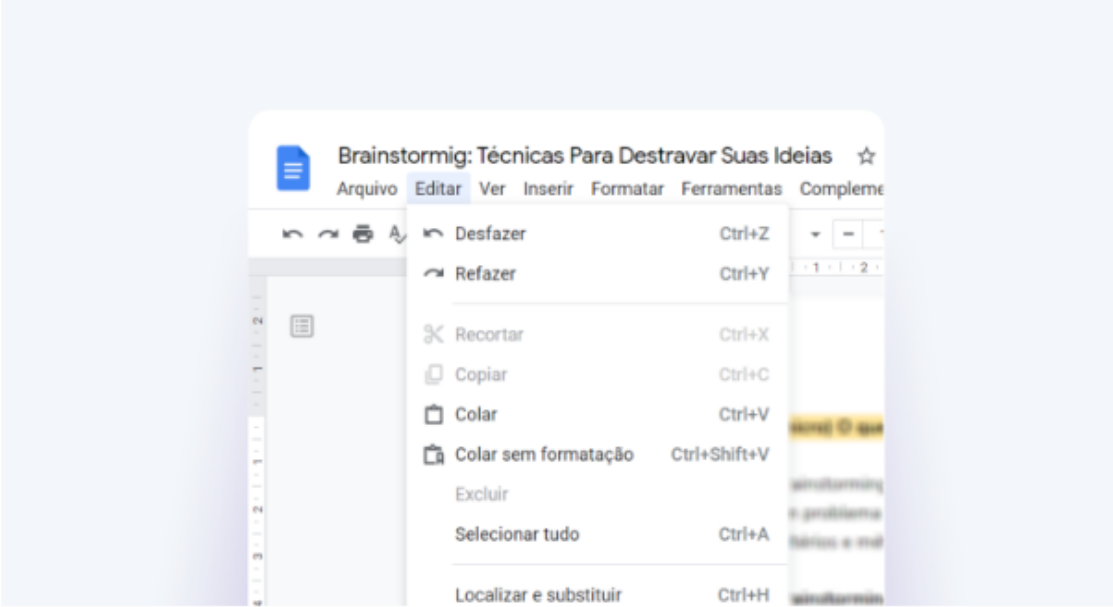
a) Controle e liberdade para o usuário
b) **Ajuda e documentação**
c) Consistência e padronização
d) Prevenção de erros

FEEDBACK: A heurística que melhor representa a imagem é a de ajuda e documentação. A imagem mostrada apresenta uma documentação de dúvidas frequentes de usuários para ajudar com relação à entrega que a empresa de comércio virtual faz.

Fonte: Elaboração própria (2022)

Figura 72 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



The image shows a screenshot of the 'Edit' menu in a Microsoft Word document titled 'Brainstormig: Técnicas Para Destruvar Suas Ideias'. The menu is open, displaying various editing options with their respective keyboard shortcuts. The options listed are: 'Desfazer' (Ctrl+Z), 'Refazer' (Ctrl+Y), 'Recortar' (Ctrl+X), 'Copiar' (Ctrl+C), 'Colar' (Ctrl+V), 'Colar sem formatação' (Ctrl+Shift+V), 'Excluir', 'Selecionar tudo' (Ctrl+A), and 'Localizar e substituir' (Ctrl+H). The 'Refazer' option is highlighted in yellow.

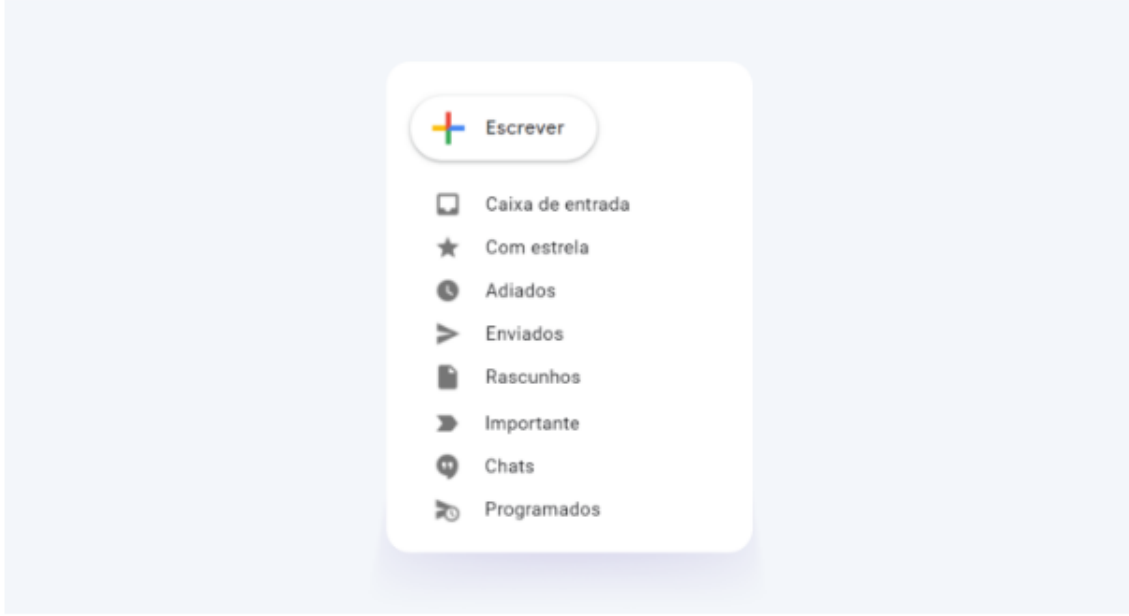
a) Consistência e padronização
b) Prevenção de erros
c) **Controle e liberdade para o usuário**
d) Reconhecimento em vez de memorização

FEEDBACK: A heurística que melhor representa a imagem é a de controle e liberdade para o usuário. O sistema apresenta botões de desfazer e refazer ações que acabam dando ao usuário o controle sobre o realizado.

Fonte: Elaboração própria (2022)

Figura 73 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



a) **Consistência e padronização**

b) Prevenção de erros

c) Reconhecimento em vez de memorização

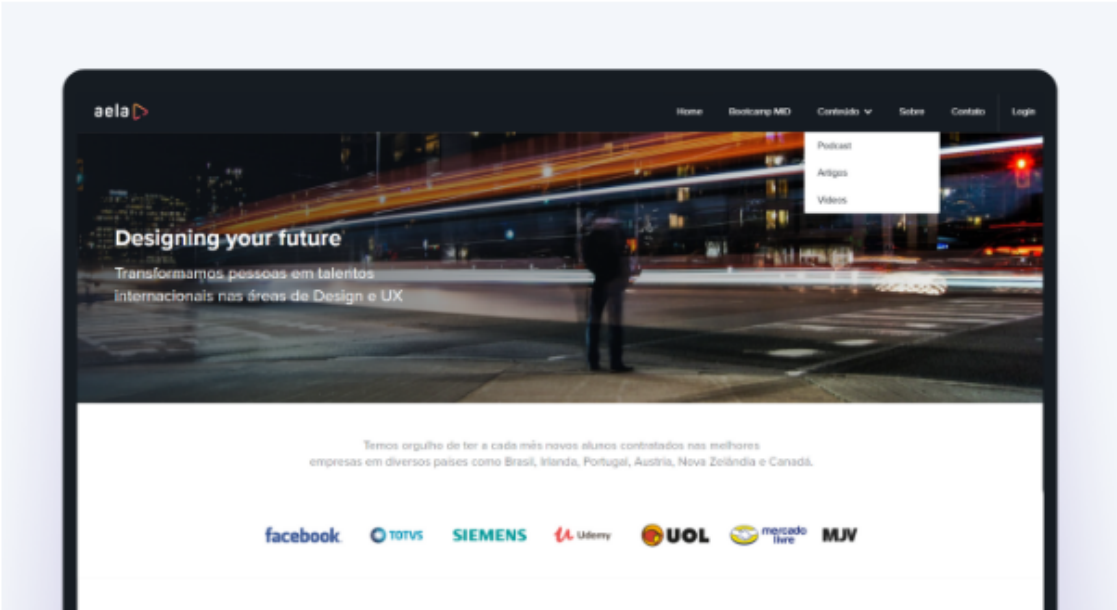
d) Eficiência e flexibilidade de uso

FEEDBACK: A heurística que melhor representa a imagem é a de consistência e padronização. O sistema mostrado possui ícones autoexplicativos para cada uma das funcionalidades e também possui um padrão de um ícone para cada funcionalidade.

Fonte: Elaboração própria (2022)

Figura 74 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



a) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

b) Ajuda e documentação

c) Visibilidade do status do sistema

d) **Estética e design minimalista**

FEEDBACK: A heurística que melhor representa a imagem é a de estética e design minimalista. O site mostrado apresenta uma estética e design bem simples e objetivo e consegue trazer todas as informações que se quer mostrar para os usuários.

Fonte: Elaboração própria (2022)

Figura 75 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



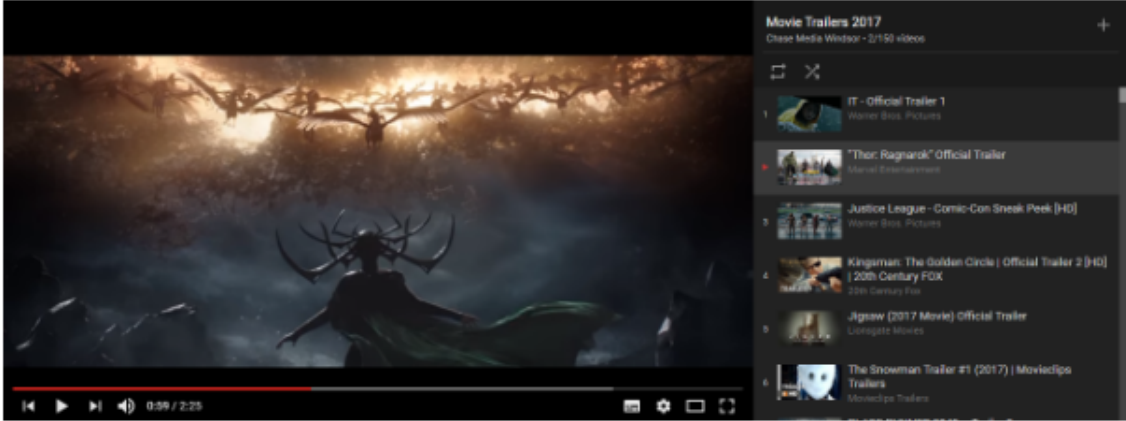
a) Controle e liberdade para o usuário
b) Consistência e padronização
c) Prevenção de erros
d) **Estética e design minimalista**

FEEDBACK: A heurística que melhor representa a imagem é a de estética e design minimalista. O aplicativo mostrado apresenta uma estética e design bem simples e objetivo e consegue trazer todas as informações que se quer mostrar para os usuários.

Fonte: Elaboração própria (2022)

Figura 76 – Pergunta do nível dois de múltipla escolha

(MÉDIO) Qual heurística melhor representa a imagem abaixo:



The image shows a video player interface. The main video area displays a dark, atmospheric scene with a figure in a green cape and a large, glowing, multi-eyed creature in the background. The video player controls at the bottom show a progress bar at 0:58 / 2:28. On the right side, there is a playlist titled 'Movie Trailers 2017' with a list of six video thumbnails and titles, including 'IT - Official Trailer 1', 'Thor: Ragnarok Official Trailer', 'Justice League - Comic-Con Sneak Peek (HD)', 'Kingsman: The Golden Circle | Official Trailer 2 (HD)', 'Jigsaw (2017 Movie) Official Trailer', and 'The Snowman Trailer #1 (2017) | Movieclips Trailers'.

a) **Visibilidade do status do sistema**

b) Prevenção de erros

c) Reconhecimento em vez de memorização

d) Eficiência e flexibilidade de uso

FEEDBACK: A heurística que melhor representa a imagem é a de visibilidade do status do sistema. A imagem mostra a barra lateral de uma plataforma de vídeo que mostra os vídeos já assistidos, o vídeo que está sendo assistido e o vídeo que será mostrado na sequência, ou seja, mostra o status da playlist de vídeos.

Fonte: Elaboração própria (2022)

D.2.2 Perguntas de verdadeiro ou falso

Figura 77 – Pergunta do nível dois de verdadeiro ou falso

<p>(MÉDIA) A heurística de visibilidade do status do sistema se refere ao sistema disponibilizar lembretes com informações úteis sobre as funcionalidades de um sistema. Essa sentença é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de reconhecimento em vez de memorização, ou seja, a afirmação estava errada. A heurística de visibilidade do status do sistema se refere à importância dos feedbacks instantâneos aos usuários, mantendo-os informados sobre o que está acontecendo no momento da interação, qual o status do momento e como ele deve conduzir seus próximos passos no processo de sua experiência.</p>
<p>(MÉDIA) A heurística de compatibilidade entre o sistema e o mundo real se refere a um padrão que a interface e linguagem de um sistema deve possuir para não confundir os usuários. Essa sentença é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de consistência e padronização, ou seja, a afirmação estava errada. A heurística de compatibilidade entre o sistema e o mundo real se refere à necessidade do sistema ter a mesma linguagem (palavras, frases, imagens e conceitos) que o usuário utiliza no dia a dia.</p>
<p>(MÉDIA) A heurística de controle e liberdade para o usuário se refere à flexibilidade de dispor informações detalhadas para os usuários poderem customizar suas ações e criar atalhos e também deve ser um sistema eficaz permitindo personalizar ações frequentes e tendo preenchimentos automáticos. Essa sentença é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de eficiência e flexibilidade, ou seja, a afirmação estava errada. A heurística de controle e liberdade para o usuário se refere à liberdade para realizar ações que o usuário deseja dentro de um sistema.</p>
<p>(MÉDIA) A heurística de consistência e padronização se refere à necessidade do sistema ter a mesma linguagem (palavras, frases, imagens e conceitos) que o usuário utiliza no dia a dia. Essa sentença é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de compatibilidade entre o sistema e o mundo real, ou seja, a afirmação estava errada. A heurística de consistência e padronização se refere a um padrão que a interface e linguagem de um sistema deve possuir para não confundir os usuários.</p>

Fonte: Elaboração própria (2022)

Figura 78 – Pergunta do nível dois de verdadeiro ou falso

(MÉDIA) A heurística de prevenção de erros se refere à importância dos feedbacks instantâneos aos usuários, mantendo-os informados sobre o que está acontecendo no momento da interação, qual o status do momento e como ele deve conduzir seus próximos passos no processo de sua experiência. Essa sentença é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de visibilidade do status do sistema, ou seja, a afirmação estava errada. A heurística de prevenção de erros se refere à prevenção de qualquer ação descuidada que um usuário possa cometer durante o uso de um sistema.

(MÉDIA) A heurística de reconhecimento em vez de memorização se refere ao sistema em casos de erros durante o uso, ajudarem os usuários a entender o problema ocorrido, além de dispor informações para poder solucionar o problema. Essa sentença é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros, ou seja, a afirmação estava errada. A heurística de reconhecimento em vez de memorização se refere ao sistema disponibilizar lembretes com informações úteis sobre as funcionalidades de um sistema.

(MÉDIA) A heurística de eficiência e flexibilidade de uso se refere à liberdade para realizar ações que o usuário deseja dentro de um sistema. Essa sentença é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de controle e liberdade para o usuário, ou seja, a afirmação estava errada. A heurística de eficiência e flexibilidade de uso se refere à flexibilidade de dispor informações detalhadas para os usuários poderem customizar suas ações e criar atalhos e também deve ser um sistema eficaz permitindo personalizar ações frequentes e tendo preenchimentos automáticos.

(MÉDIA) A heurística de estética e design minimalista se refere ao sistema em casos de erros durante o uso, ajudarem os usuários a entender o problema ocorrido, além de dispor informações para poder solucionar o problema. Essa sentença é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros, ou seja, a afirmação estava errada. A heurística de estética e design minimalista se refere ao sistema possuir elementos visuais diretos e simples, sem o uso em excesso ou desnecessário de elementos.

Fonte: Elaboração própria (2022)

Figura 79 – Pergunta do nível dois de verdadeiro ou falso

(MÉDIA) A heurística de ajuda os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros se refere à prevenção de qualquer ação descuidada que um usuário possa cometer durante o uso de um sistema. Essa sentença é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de prevenção de erros, ou seja, a afirmação estava errada. A heurística de ajuda os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros se refere ao sistema em casos de erros durante o uso, ajudarem os usuários a entender o problema ocorrido, além de dispor informações para poder solucionar o problema.

(MÉDIA) A heurística de ajuda e documentação se refere ao sistema possuir elementos visuais diretos e simples, sem o uso em excesso ou desnecessário de elementos. Essa sentença é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de estética e design minimalista, ou seja, a afirmação estava errada. A heurística de ajuda e documentação se refere aos sistemas possuírem uma seção de ajuda para os usuários poderem obter respostas rápidas.


Fonte: Elaboração própria (2022)

D.3 Perguntas do nível três

D.3.1 Perguntas de múltipla escolha

Figura 80 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:



Como o nosso **Pão Francês** é chamado em diferentes regiões do Brasil

- a) Consistência e padronização
- b) Controle e liberdade para o usuário
- c) **Compatibilidade entre o sistema e o mundo real**
- d) Prevenção de erros

FEEDBACK: A heurística que melhor representa a imagem é a de compatibilidade entre o sistema e o mundo real. A imagem mostra como o pão francês é chamado em diferentes regiões do Brasil, ou seja, como a palavra se adapta de um lugar para o outro para que a maioria do público de determinada região entenda do que se trata facilmente. O mesmo serve para um sistema, que adaptando a linguagem utilizada aumenta a compatibilidade.

Fonte: Elaboração própria (2022)

Figura 81 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:



a) **Consistência e padronização**

b) Prevenção de erros

c) Reconhecimento em vez de memorização

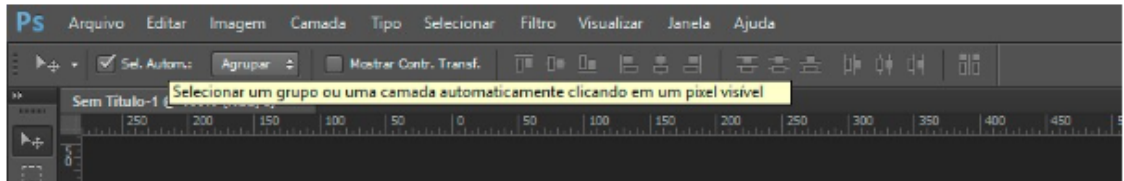
d) Eficiência e flexibilidade de uso

FEEDBACK: A heurística que melhor representa a imagem é a de consistência e padronização. O site mostrado preserva um padrão de cabeçalho que diversos sites utilizam, o que acaba ficando padronizado com diversos outros sites e facilitando a vida dos usuários.

Fonte: Elaboração própria (2022)

Figura 82 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:



a) Eficiência e flexibilidade de uso

b) **Reconhecimento em vez de memorização**

c) Estética e design minimalista


d) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

FEEDBACK: A heurística que melhor representa a imagem é a de reconhecimento em vez de memorização. A imagem mostra um lembrete mostrado para o usuário ao passar o mouse sobre alguma das opções clicáveis e nesse caso é uma funcionalidade que reconhece todas as camadas no editor de imagens para que facilite a edição da imagem.

Fonte: Elaboração própria (2022)

Figura 83 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:




a) Reconhecimento em vez de memorização
b) **Compatibilidade entre o sistema e o mundo real**
c) Eficiência e flexibilidade de uso
d) Estética e design minimalista

FEEDBACK: A heurística que melhor representa a imagem é a de compatibilidade entre o sistema e o mundo real. O serviço de edição de fotos mostrado faz o uso de elementos visuais que têm funções no sistema exatamente dos objetos físicos. Há aqui uma compatibilidade entre as funcionalidades dos elementos com a versão física deles.

Fonte: Elaboração própria (2022)

Figura 84 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:



a) **Consistência e padronização**

b) Estética e design minimalista

c) Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros

d) Ajuda e documentação

FEEDBACK: A heurística que melhor representa a imagem é a de consistência e padronização. A imagem mostra um padrão nos botões de responder e cancelar que diversos sites utilizam, o que acaba ficando padronizado com diversos outros sites e facilitando a vida dos usuários.

Fonte: Elaboração própria (2022)

Figura 85 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:



a) Compatibilidade entre o sistema e o mundo real

b) Visibilidade do status do sistema

c) **Reconhecimento em vez de memorização**

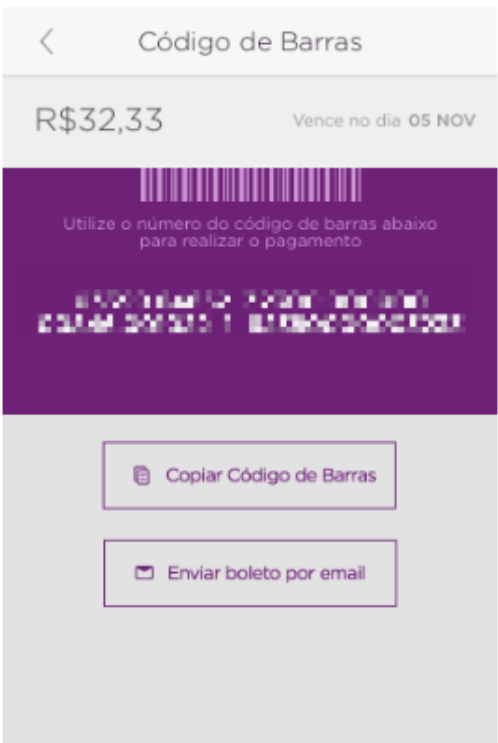
d) Controle e liberdade para o usuário

FEEDBACK: A heurística que melhor representa a imagem é a de reconhecimento em vez de memorização. No site mostrado o caminho realizado pelo usuário até chegar na página mostrada está localizada no canto superior esquerdo, e isso auxilia o usuário a se lembrar do caminho feito da página inicial até a atual.

Fonte: Elaboração própria (2022)

Figura 86 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:




a) Consistência e padronização
b) Controle e liberdade para o usuário
c) **Estética e design minimalista**
d) Prevenção de erros

FEEDBACK: A heurística que melhor representa a imagem é a de estética e design minimalista. O aplicativo mostrado apresenta uma estética e design bem simples e objetivo e consegue trazer todas as informações que se quer mostrar para os usuários.

Fonte: Elaboração própria (2022)

Figura 87 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:




Número do cartão

Data de validade

Mês Ano

Código de segurança



Número de parcelas

a) Ajuda e documentação

b) Prevenção de erros

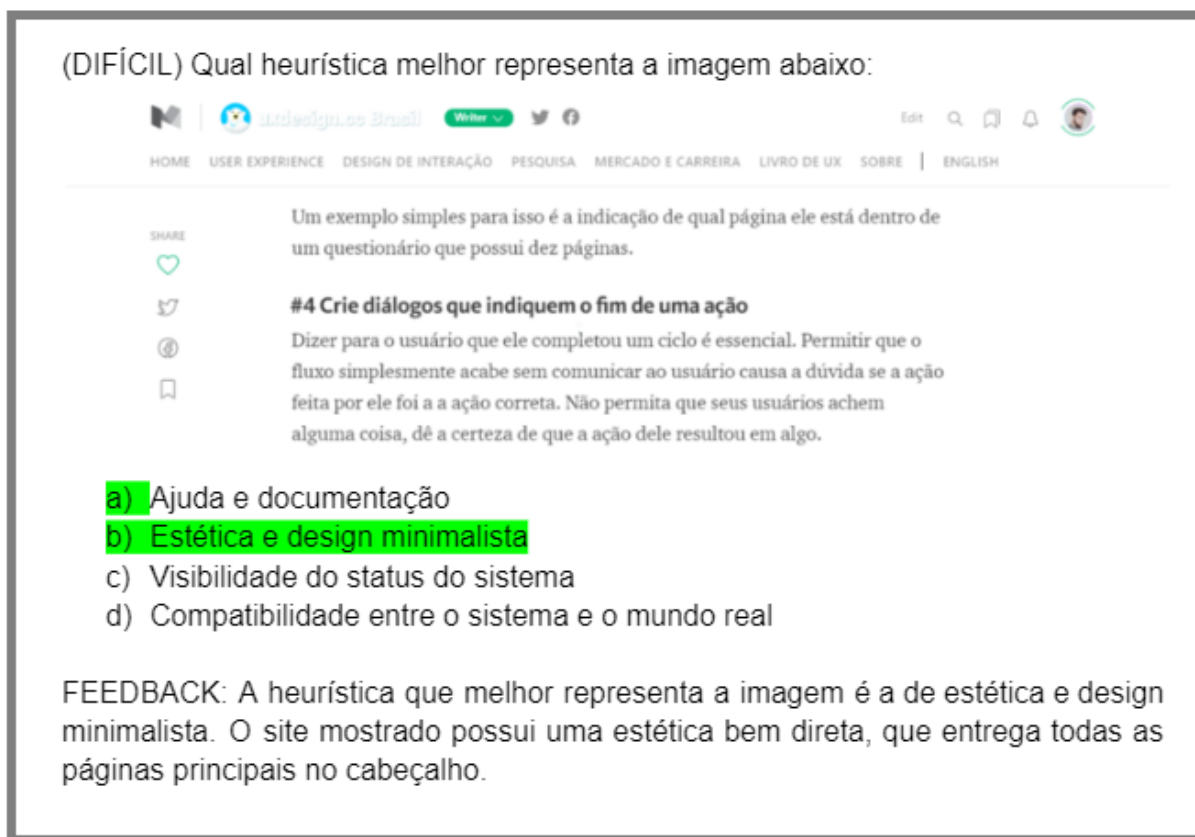
c) Reconhecimento em vez de memorização

d) Eficiência e flexibilidade de uso

FEEDBACK: A heurística que melhor representa a imagem é a de ajuda e documentação. A imagem mostrada apresenta uma sutil ajuda ao usuário na imagem que ilustra o campo código de segurança, mostrando o local no cartão que o usuário achará o código.

Fonte: Elaboração própria (2022)

Figura 88 – Pergunta do nível três de múltipla escolha



(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:

Um exemplo simples para isso é a indicação de qual página ele está dentro de um questionário que possui dez páginas.

#4 Crie diálogos que indiquem o fim de uma ação

Dizer para o usuário que ele completou um ciclo é essencial. Permitir que o fluxo simplesmente acabe sem comunicar ao usuário causa a dúvida se a ação feita por ele foi a ação correta. Não permita que seus usuários achem alguma coisa, dê a certeza de que a ação dele resultou em algo.

a) Ajuda e documentação

b) Estética e design minimalista

c) Visibilidade do status do sistema


d) Compatibilidade entre o sistema e o mundo real

FEEDBACK: A heurística que melhor representa a imagem é a de estética e design minimalista. O site mostrado possui uma estética bem direta, que entrega todas as páginas principais no cabeçalho.

Fonte: Elaboração própria (2022)

Figura 89 – Pergunta do nível três de múltipla escolha

(DIFÍCIL) Qual heurística melhor representa a imagem abaixo:

An illustration of a person with a magnifying glass looking at a large '404' error message. The person is wearing a red shirt and a blue hat. The '404' is written in large, blue, stylized numbers. The background is light blue with some green leaves and a small blue circle.

a) Ajuda e documentação
b) Visibilidade do status do sistema
c) Controle e liberdade para o usuário
d) **Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros**

FEEDBACK: A heurística que melhor representa a imagem é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros. O sistema quando a página não é encontrada mostra para o usuário o erro 404 para que possa ser visto que um erro ocorreu e poderem voltar para a página anterior.

Fonte: Elaboração própria (2022)

D.3.2 Perguntas de verdadeiro ou falso

Figura 90 – Pergunta do nível três de verdadeiro ou falso

<p>(DIFÍCIL) A empresa Mayflower, de origem inglesa, chegou ao Brasil com sua nova rede social, mas não foi traduzida para o português e com isso vários usuários brasileiros reclamaram com a empresa sobre a rede social não estar em português. Nesse caso faltou à empresa Mayflower aplicar a heurística de compatibilidade entre o sistema e o mundo real. Essa afirmação é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de compatibilidade entre o sistema e o mundo real, ou seja, a afirmação estava correta.</p>
<p>(DIFÍCIL) O material design da google é um ótimo exemplo da aplicação da heurística de consistência e padronização. Essa afirmação é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de consistência e padronização, ou seja, a afirmação estava correta.</p>
<p>(DIFÍCIL) Quando estamos acessando sites que possuem login e nosso navegador nos pergunta se queremos salvar nosso usuário e senha, esse é um caso da aplicação da heurística de eficiência e flexibilidade de uso. Essa afirmação é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de reconhecimento em vez de memorização, ou seja, a afirmação estava errada. Apesar de se ter um momento de memorização da senha, o usuário pode ir posteriormente nas configurações de seu navegador e visualizar a senha salva nos sistemas, precisando digitar a senha de seu usuário do computador.</p>
<p>(DIFÍCIL) Quando você, enquanto usuário do sistema operacional windows, resolve dar os comandos de “Control + C”, “Control + V” ou então “Alt + Tab”, esses são comandos que mostram a implementação da heurística de eficiência e flexibilidade de uso. Essa afirmação é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de eficiência e flexibilidade de uso, ou seja, a afirmação estava correta.</p>

Fonte: Elaboração própria (2022)

Figura 91 – Pergunta do nível três de verdadeiro ou falso

<p>(DIFÍCIL) A empresa <u>Takeaway</u> implementou um sistema de suporte via chat em tempo real para resolver problemas de seus clientes. A empresa acabou de implementar uma funcionalidade que faz uso da heurística de ajuda e documentação. Essa afirmação é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de ajuda e documentação, ou seja, a afirmação estava correta.</p>
<p>(DIFÍCIL) No processo de fechamento de uma compra em uma loja virtual a heurística mais importante de se ter implementada é a de estética e design minimalista. Essa afirmação é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de visibilidade do status do sistema, ou seja, a afirmação estava errada. Em uma loja virtual no processo de finalização de uma compra o que mais interessa ao usuário no momento é saber em que estágio da compra ele se encontra e ao final saber se a compra foi concluída com sucesso ou não, e isso se dá por meio da visibilidade do status.</p>
<p>(DIFÍCIL) Quando se está fazendo compras em uma loja virtual é importante que os usuários possam adicionar e excluir produtos de seus carrinhos no momento que eles quiserem até o pagamento ser realizado e o carrinho ser fechado. Quando uma empresa implementa um carrinho de compras em seu comércio virtual seguindo esses moldes, é um claro exemplo da heurística de consistência e padronização sendo seguido. Essa afirmação é verdadeira ou falsa? Falso</p> <p>FEEDBACK: A heurística da afirmação é a de controle e liberdade para o usuário, ou seja, a afirmação estava errada. Quando damos esse poder de escolha para o usuário adicionar ou excluir itens do carrinho estamos dando controle e liberdade para o mesmo.</p>
<p>(DIFÍCIL) Alguns comércios virtuais começaram a implementar a compra com 1 clique. Essa funcionalidade é uma aplicação da heurística de eficiência e flexibilidade de uso. Essa afirmação é verdadeira ou falsa? Verdadeiro</p> <p>FEEDBACK: A heurística da afirmação é a de eficiência e flexibilidade de uso, ou seja, a afirmação estava correta.</p>

Fonte: Elaboração própria (2022)

Figura 92 – Pergunta do nível três de verdadeiro ou falso

(DIFÍCIL) A plataforma de vídeos online VocêTV possui uma barra lateral informando qual vídeo está sendo assistido, quais já foram assistidos e quais são os próximos a serem exibidos. A plataforma VocêTV faz uso da heurística de reconhecimento em vez de memorização. Essa afirmação é verdadeira ou falsa?

Falso

FEEDBACK: A heurística da afirmação é a de visibilidade do status do sistema, ou seja, a afirmação estava errada. Mostrando os vídeos assistidos, o que se está assistindo e o que será assistido, a plataforma mostra o status da playlist de vídeos.

(DIFÍCIL) Quando você tenta acessar uma página de um site e ele te mostrar uma página com o erro 404, mostra que o site implementou a heurística de ajuda e documentação. Essa afirmação é verdadeira ou falsa? **Falso**

FEEDBACK: A heurística da afirmação é a de ajuda aos usuários a reconhecerem, diagnosticarem e recuperarem-se de erros, ou seja, a afirmação estava errada. O erro 404 ajuda o usuário a saber que teve um erro e poder voltar à página anterior.

Fonte: Elaboração própria (2022)

Anexos

ANEXO A – Modelo MEEGA+

Tabela 17 – Itens de instrumento de medição MEEGA+

Fator de qualidade	Dimensão	ID	Descrição
Usabilidade	Estética	1	O design do jogo é atraente (interface, gráficos, cartões, placas, etc.).
		2	A fonte e as cores do texto são bem combinadas e consistentes.
	Aprendizagem	3	Eu precisava aprender algumas coisas antes de poder jogar.
		4	Aprender a jogar esse jogo foi fácil para mim.
		5	Acho que a maioria das pessoas aprenderia a jogar esse jogo muito rapidamente.
	Operabilidade	6	Acho que o jogo é fácil de jogar.
		7	As regras do jogo são claras e fáceis de entender.
	Acessibilidade	8	As fontes (tamanho e estilo) usadas no jogo são fáceis de ler.
		9	As cores usadas no jogo são significativas.
Experiência Do Jogador	Confiança	10	O conteúdo e a estrutura me ajudaram a ter certeza de que aprenderia com este jogo.
	Desafio	11	Este jogo é apropriadamente desafiador para mim.
		12	O jogo oferece novos desafios (oferece novos obstáculos, situações ou variações) em um ritmo apropriado.
		13	O jogo não se torna monótono à medida que avança (tarefas repetitivas ou enfadonhas).
	Satisfação	14	Concluir as tarefas do jogo me deu uma sensação satisfatória de realização.
		15	É pelo meu esforço pessoal que consegui avançar no jogo.
		16	Estou satisfeito com as coisas que aprendi com o jogo.
	Interação Social	17	Eu recomendaria este jogo aos meus colegas.
		18	Pude interagir com outros jogadores durante o jogo.
		19	O jogo promove a cooperação e / ou competição entre os jogadores.
	Diversão	20	Me senti bem interagindo com outros jogadores durante o jogo.
		21	Eu me diverti com o jogo.
		22	Algo aconteceu durante o jogo (elementos do jogo, competição, etc.) que me fez sorrir.
	Atenção Focada	23	Havia algo interessante no início do jogo que chamou minha atenção.
		24	Eu estava tão envolvido em minhas tarefas de jogo que perdi a noção do tempo.
25		Esqueci meu ambiente imediato enquanto jogava.	
Relevância	26	O conteúdo do jogo é relevante para os meus interesses.	
	27	É claro para mim como o conteúdo do jogo se relaciona com o curso.	
	28	Este jogo é um método de ensino adequado para este curso.	
Aprendizagem Percebida	29	Prefiro aprender com este jogo a aprender de outras maneiras (por exemplo, outros métodos de ensino).	
	30	O jogo contribuiu para o meu aprendizado neste curso.	
	31	O jogo permitiu um aprendizado eficiente em comparação com outras atividades do curso.	

Fonte: Adaptado de (PETRI; WANGENHEIM, 2019)


ANEXO B – Personas utilizadas no projeto

Figura 93 – Persona primária

Identidade	
	<p>Nome: Victor Matheus Farias Idade: 19 anos Ocupação: Estudante de Engenharia de Software na UnB, Faculdade do Gama.</p>
<p>Fonte: Pixabay¹</p>	
Descrição	
<p>Aprender um conteúdo novo é meu o principal objetivo ao usar jogos educacionais. Atualmente eu uso alguns, mas com uma frequência moderada, não gosto de gastar muito tempo com jogos. Vejo-os como ferramentas que me auxiliam no processo de aprendizagem. Estou cursando a disciplina de IHC e não tenho muito conhecimento em relação a elaboração de design de interfaces e o pouco que tenho está somente no âmbito disciplinar. Desejaria utilizar um jogo que me ajudasse a aprender o conteúdo.</p> <p>Geralmente quando vou estudar ou sanar alguma dúvida que tenho sobre o conteúdo eu pesquiso na internet, em alguns casos assisto vídeo aulas e também utilizo do material disponibilizado pelo professor.</p> <p>Um jogo educacional pra mim, deve responder bem às minhas ações, me ensinando caso eu erre por exemplo; deve ter um design legal, talvez com um tema; deve ser simples de se aprender a jogar, não tendo regras extensas e tutorias longos; e também deve ser fácil de jogar.</p> <p>E por fim o que eu espero de um jogo educacional é sentir satisfação, aprender com um pouco de desafio e diversão é muito bom. Ter a percepção da relevância do assunto que estou aprendendo é algo que me dá confiança que vou atingir meu objetivo de estudo. E mesmo não gostando de gastar muito tempo com jogos, mas se eu sentir que estou sendo produtivo com certeza vou me manter focado.</p>	


Fonte: Silva (2021)

Figura 94 – Persona secundária

Identidade	
	<p>Nome: Afonso Souza de Queiroz Idade: 19 anos Ocupação: Estudante de Engenharia de Software na UnB, Faculdade do Gama</p>
Fonte: Pixabay ²	
Descrição	
<p>Já joguei alguns jogos educacionais onde meu principal objetivo era aprender um conteúdo novo. Além disso era interessante quando o jogo possibilitava que eu revisasse o conteúdo e avaliasse o que tinha aprendido. Eu jogava com certa moderação. Por mais que atualmente eu não esteja usando algum jogo educacional, eu parei de jogar pois alcancei meus objetivos de estudo e me sentiria da mesma forma se houvesse algum jogo que me ajudasse nas matérias da faculdade.</p> <p>Ainda não fiz a disciplina de IHC, por isso não tenho muito conhecimento em relação a elaboração de design de interfaces, sendo que o pouco que tenho veio de projetos de outras disciplinas e atividades extra curriculares.</p> <p>Gosto mais de estudar sozinho, pesquisando na internet e indo por conta própria atrás de materiais de estudo. Apenas quando não acho nada relevante, recorro aos materiais disponibilizados pelo professor.</p> <p>Para mim, um jogo onde vou aprender deve ter um design com um padrão simples, mas que desperte interesse; deve me dar bons feedbacks a cada interação que faço no jogo; deve ser algo simples de aprender como jogar e simples de jogar; e caso tenha textos sobre o conteúdo, seria bom que os textos fossem bem objetivos e com uma cor e fonte que facilitasse a leitura.</p> <p>Espero ter a sensação logo de cara que não vou perder meu tempo com o jogo, mas que ao final vou perceber o quanto foi importante aprender o conteúdo e o quanto foi satisfatório ter me divertido aprendendo. E espero também de certa forma ser desafiado, pois isso me ajuda a focar na atividade que estou realizando.</p>	

Fonte: Silva (2021)

Figura 95 – Persona suplementar

Identidade	
	<p>Nome: Natália Figueiredo Idade: 23 anos Ocupação: Estudante de Engenharia de Software na UnB, Faculdade do Gama.</p>
Fonte: Pixabay ³	
Descrição	
<p>Nunca joguei jogos educacionais, pois não conheci nenhum que tinha o propósito de ensinar o que eu desejava. No caso se eu encontrasse um jogo o qual eu pudesse aprender um conteúdo novo e pudesse revisá-lo quando necessário seria interessante. Estou fazendo a disciplina de IHC e desejaria utilizar um jogo que me ajudasse a aprender o conteúdo, pois não tenho muito conhecimento em relação a elaboração de design de interfaces e o pouco que tenho foi somente a partir da disciplina de IHC.</p> <p>Geralmente quando vou estudar ou sanar alguma dúvida que tenho sobre o conteúdo eu pesquiso na internet. Em ocasiões bem específicas eu assisto vídeo aulas e também utilizo do material disponibilizado pelo professor.</p> <p>Acho que um jogo para se estudar teria de ter um design simples, uma lógica de jogo e regras fáceis de se lembrar. O jogo não deveria ser muito difícil, sendo que o objetivo com ele é aprender. Talvez recompensas e mensagens evidenciando meu progresso seriam interessantes.</p> <p>Ao jogar, espero encontrar desafios não muito difíceis, mas que despertem minha atenção. Seria bom, perceber logo de início que o jogo traz um conteúdo relevante e que vou conseguir aprendê-lo. E por fim não seria nada ruim sentir prazer em aprender e ainda me divertir.</p>	

Fonte: Silva (2021)

Figura 96 – Anti persona

Identidade	
	<p>Nome: Rafael Medeiros Idade: 28 anos Ocupação: Estudante de Educação Física na UnB, campus Darcy Ribeiro.</p>
Fonte: Pixabay ⁴	
Descrição	
<p>Não tenho o costume de usar jogos educacionais, devo ter jogado uma vez ou outra, mas não me interessei muito e rapidamente o jogo se tornou monótono pra mim. Gosto mais de jogos esportivos como futebol e basquete. Em relação ao conhecimento acadêmico, me concentro apenas nas disciplinas do meu curso.</p> <p>No geral, sou bem competitivo, dou o sangue para conquistar um gol ou fazer um ponto, e fico mais ainda quando se trata de campeonatos, luto pra ficar em primeiro. Em jogos digitais eu gosto de ser envolvido com uma história bacana e fico bastante empolgado com gráficos bem realistas e sofisticados.</p> <p>Sou mais fã de jogos com trabalho em equipe do que jogos individuais. Gosto tanto atividades competitivas e lúdicas que se desse pra aprender com elas eu com certeza trocaria qualquer livro pra aprender me divertindo.</p> <p>Como futuro profissional da área de Educação Física meu foco está no estudo de formas de estender o esporte e jogos para públicos com necessidades especiais.</p>	

Fonte: Silva (2021)