



UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação
Departamento de Ciências de Computação

Design, Desenvolvimento e Avaliação do Jogo de Narrativa
Invertida para a Plataforma RUFUS

Tiago Marino Silva



São Carlos – SP

Design, Desenvolvimento e Avaliação do Jogo de Narrativa Invertida para a Plataforma RUFUS

Tiago Marino Silva

Orientadora: Profa. Dra. Kamila Rios da Hora Rodrigues

Monografia final de conclusão de curso apresentada
ao Instituto de Ciências Matemáticas e de
Computação – ICMC-USP, como requisito parcial
para obtenção do título de Bacharel em Computação.
Área de Concentração: Sistemas Computacionais

USP – São Carlos
Novembro de 2022

Silva, Tiago Marino

Design, Desenvolvimento e Avaliação do Jogo de Narrativa Invertida para a Plataforma RUFUS / Tiago Marino Silva. - São Carlos - SP, 2022.

54 p.; 29,7 cm.

Orientadora: Kamila Rios da Hora Rodrigues.

Monografia (Graduação) - Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos - SP, 2022.

1. Plataforma RUFUS. 2. Jogos Digitais Sérios.
3. Design. 4. Avaliação. I. Rodrigues, Kamila Rios da Hora. II. Instituto de Ciências Matemáticas e de Computação (ICMC/USP). III. Título.

RESUMO

SILVA, T. M.. **Design, Desenvolvimento e Avaliação do Jogo de Narrativa Invertida para a Plataforma RUFUS**. 2022. 54 f. Monografia (Graduação) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

Este trabalho descreve o design, desenvolvimento e avaliação da mecânica de jogo denominada Narrativa Invertida a ser disponibilizada na plataforma de autoria de jogos sérios RUFUS. A RUFUS é uma ferramenta que auxilia profissionais de saúde no tratamento de seus pacientes, permitindo que eles criem e configurem jogos terapêuticos para serem usados com esses. A plataforma disponibiliza diferentes gêneros de jogos, denominados mecânicas. A narrativa invertida, na qual o jogador quem constrói a sua história, permite maior expressão criativa dos usuários e maior diálogo com temas sensíveis, como a morte.

Palavras-chave: Plataforma RUFUS, Jogos Digitais Sérios, Design, Avaliação.

ABSTRACT

SILVA, T. M.. **Design, Desenvolvimento e Avaliação do Jogo de Narrativa Invertida para a Plataforma RUFUS**. 2022. 54 f. Monografia (Graduação) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

This work describes the design, development and evaluation of the game mechanic entitled as Inverted Storytelling, that will be available at the serious games authoring platform RUFUS. RUFUS is a tool that helps health professionals in the treatment of their patients, allowing them to create and configure therapeutic games to be used with them. The platform offers different game genres, denominated as mechanics. The Inverted Storytelling, in which the player builds their own story, allows for a greater creative expression of users and greater dialogue about sensitive topics, such as death.

Keywords: RUFUS Platform, Digital Serious Games, Design, Evaluation.

LISTA DE ILUSTRAÇÕES

Figura 1 – SAM (BRADLEY; LANG, 1994).	23
Figura 2 – Protótipo da Designer - Criação de Personagem.	26
Figura 3 – Exemplo de tela do protótipo - Personagens.	26
Figura 4 – Exemplo de tela do protótipo - Balões.	26
Figura 5 – Criação de Personagem na Mecânica de Narrativa.	27
Figura 6 – Menu de Visualização de Páginas.	27
Figura 7 – Tela inicial - Protótipo de média fidelidade da Designer	29
Figura 8 – Tela inicial - Protótipo da aplicação Móvel.	29
Figura 9 – Fluxo de criação de uma narrativa.	30
Figura 10 – Seleção de Fundo - Protótipo de média fidelidade da designer.	30
Figura 11 – Seleção de Fundo - Protótipo da aplicação móvel.	31
Figura 12 – Adicionando Personagens - Protótipo de média fidelidade elaborado pela designer.	31
Figura 13 – Arrastando Personagem para a Página.	32
Figura 14 – Personagens com Diferentes Posições, Tamanhos e Orientações.	32
Figura 15 – Adicionando Balão de Fala - Protótipo de média fidelidade elaborado pela designer.	33
Figura 16 – Adicionando Objetos - Protótipo da aplicação móvel	33
Figura 17 – Adicionando Balão de Fala - Protótipo da aplicação móvel.	33
Figura 18 – Balões de Fala com Texto.	34
Figura 19 – Nova Página Montada.	34
Figura 20 – Menu de Remoção - Protótipo de média fidelidade elaborado pela designer.	35
Figura 21 – Menu de Remoção - Protótipo da aplicação móvel.	35
Figura 22 – Tipos de jogos que os voluntários estão habituados a jogar.	37
Figura 23 – Correspondência entre o sistema e o mundo real.	39
Figura 24 – Controle e liberdade para o usuário.	40
Figura 25 – Consistência e padronização.	41
Figura 26 – Reconhecimento ao invés de memorização - questões negativas.	42
Figura 27 – Reconhecimento ao invés de memorização - questões positivas.	42
Figura 28 – Flexibilidade e eficiência.	43
Figura 29 – Estética e design minimalista.	44
Figura 30 – Satisfação.	45
Figura 31 – Motivação.	45

Figura 32 – Sentimento de Controle.	46
Figura 33 – Elementos Sobrepostos.	47

LISTA DE TABELAS

Tabela 1 – <i>Score</i> dos Usuários	38
Tabela 2 – Média e Desvio Padrão dos Domínios do SAM.	46

LISTA DE QUADROS

Quadro 1 – Perfil de usuários voluntários.	36
Quadro 2 – Tarefas da Interação com Roteiro	37
Quadro 3 – Perguntas Feitas Durante Entrevista.	46

LISTA DE ABREVIATURAS E SIGLAS

DP	Design Participativo
EUD	<i>End-User Development</i>
FoG	Fellowship of the Game
POO	Programação Orientada a Objeto
PRCEU	..	Pró-Reitoria de Cultura e Extensão Universitária
SAM	<i>Self-Assessment Manikin</i>
SUS	<i>System Usability Scale</i>
UGL	<i>Usp Game Link</i>

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Motivação e Contextualização	17
1.2	Objetivos	18
1.3	Objetivos Específicos	18
1.4	Organização do Documento	18
2	METODOLOGIAS EMPREGADAS	19
2.1	Etapa de Design	19
2.1.1	<i>Design Participativo</i>	19
2.2	Etapa de Desenvolvimento	20
2.2.1	<i>Prototipagem Evolutiva</i>	20
2.2.2	<i>Scrum</i>	20
2.2.3	<i>Programação em Pares</i>	20
2.2.4	<i>Revisão de Código</i>	21
2.2.5	<i>Ferramentas Utilizadas</i>	21
2.3	Etapa de Avaliação	21
2.3.1	<i>System Usability Scale (SUS)</i>	22
2.3.1.1	<i>Heurísticas de Nielsen</i>	22
2.3.2	<i>Self-Assesment Manikin (SAM)</i>	23
3	DESENVOLVIMENTO DA NARRATIVA INVERTIDA - APLICAÇÃO MÓVEL	25
3.1	Atividades Realizadas	25
3.1.1	<i>Etapa de Design</i>	25
3.1.1.1	<i>Fluxo de Execução da Narrativa Invertida</i>	26
3.1.2	<i>Etapa de Desenvolvimento</i>	28
3.1.2.1	<i>Implementação da Aplicação Móvel</i>	28
3.1.3	<i>Etapa de Avaliação</i>	35
3.1.3.1	<i>Testes de Usabilidade</i>	36
3.1.3.2	<i>Interação com Roteiro</i>	37
3.1.3.3	<i>Resultados do System Usability Scale (SUS)</i>	38
3.1.3.4	<i>Resultados do Self-Assessment Manikin (SAM)</i>	45
3.1.3.5	<i>Entrevista Semiestruturada</i>	46

3.1.3.6	<i>Problemas de Usabilidade Encontrados</i>	47
3.1.3.7	<i>Melhorias a serem realizadas</i>	48
4	CONCLUSÃO	49
4.1	Contribuições do Trabalho	49
4.2	Dificuldades e Limitações do Trabalho	49
4.3	Considerações Sobre o Curso de Graduação	50
	REFERÊNCIAS	53

INTRODUÇÃO

1.1 Motivação e Contextualização

Jogos digitais são sistemas de software usados para o lazer, entretenimento e como ferramentas para a saúde e educação (GRAMMENOS; SAVIDIS; STEPHANIDIS, 2009; YUAN; FOLMER; HARRIS, 2011). Ao se tratar do uso de jogos digitais por profissionais da saúde, a fim de permitir que seus pacientes expressem seus sentimentos durante a terapia e melhorem suas habilidades e relações interpessoais, esses são classificados como jogos terapêuticos (CHEUNG, 2006). A construção de jogos terapêuticos pode estar envolvida em um cenário complexo, o qual requer uma equipe multidisciplinar e uma visão abrangente. Neste sentido, foi criada a plataforma RUFUS (RODRIGUES *et al.*, 2021)¹, uma plataforma que permite a autoria de jogos terapêuticos por parte dos profissionais de saúde. Essa plataforma consiste em uma aplicação web, na qual é feita a autoria e/ou criação dos jogos, e um aplicativo móvel, o qual faz a leitura do jogo criado e o executa para o jogador. Atualmente, a plataforma possui quatro mecânicas que podem ser usadas na criação dos jogos, são elas: *quiz*, encaixe, plataforma ou coleta e narrativa (*storytelling*) (VERHALEN *et al.*, 2022).

Durante o estudo de Verhalen *et al.* (2022) sobre jogos para apoiar o diálogo sobre a morte com crianças, observou-se a necessidade de adicionar à plataforma uma mecânica que permitisse aos usuários se expressarem de forma mais livre. A pesquisa de Verhalen *et al.*, especificamente, precisava viabilizar que profissionais da área de saúde mental pudessem acompanhar pacientes em estágio de luto. Desta necessidade, foi realizado o design da mecânica denominada de Narrativa Invertida, em que o jogador constrói a sua própria narrativa a partir de elementos pré definidos. Tal mecânica será explicada na Seção 1.2.

Este trabalho de conclusão de curso é parte de um projeto de pesquisa mais amplo da orientadora, financiado pela Pró-Reitoria de Cultura e Extensão Universitária (PRCEU) da USP, que estuda a área de *End-User Development* (EUD). EUD é definido como um conjunto de métodos, técnicas e ferramentas que permitem que usuários de sistemas de computadores, que estão no papel não profissional de desenvolvedor de *software*, em algum ponto, possa criar, modificar ou expandir os artefatos que compõem esse sistema (LIEBERMAN *et al.*, 2006).

¹ <https://rufus.icmc.usp.br>

1.2 Objetivos

O objetivo principal deste trabalho foi realizar o design, o desenvolvimento e a avaliação da nova mecânica, Narrativa Invertida, para a plataforma RUFUS (interface para dispositivos móveis).

A mecânica nova se baseia em livros de história com adesivos, nos quais a criança pode construir algumas partes da história desenhando e colando os adesivos na folha em branco. Adaptando essa ideia ao formato digital e à plataforma RUFUS, o profissional da saúde escolhe os *assets*² que serão disponibilizados ao seu paciente, separados em grupos como: objetos, personagens, cenários e outros. Então, o paciente pode montar uma história sequencial, posicionando esses *assets* de forma livre em cada página e criando novas páginas.

1.3 Objetivos Específicos

Os objetivos específicos deste trabalho incluíram:

- Coletar requisitos, junto à equipe do projeto, que respaldassem a criação da nova mecânica;
- Realizar o design da mecânica de Narrativa Invertida em conjunto com a designer do projeto;
- Conduzir o desenvolvimento da mecânica de Narrativa Invertida na aplicação móvel da RUFUS, de maneira iterativa e incremental;
- Coordenar a equipe de desenvolvimento da aplicação móvel para realização das tarefas de desenvolvimento;
- Conduzir atividades de avaliação da mecânica de Narrativa Invertida.

1.4 Organização do Documento

O restante desta monografia está organizada da seguinte maneira: o Capítulo 2 descreve os métodos usados no design, desenvolvimento e avaliação da mecânica, o Capítulo 3 descreve a condução das etapas de design, desenvolvimento e avaliação, assim como os resultados obtidos com tal avaliação e, o Capítulo 4 por fim, discorre sobre as considerações finais do trabalho e sobre o curso de graduação.

² Elementos que constituem um jogo.

METODOLOGIAS EMPREGADAS

Durante as etapas de design e desenvolvimento deste trabalho, foram empregadas práticas de Design Participativo (DP) (BARBOSA; SILVA, 2010), como *brainstorming* e prototipagem em conjunto com diferentes partes interessadas, somadas à Prototipagem Evolutiva (PRESSMMAN, 2011) da Engenharia de Software. Tanto o DP quanto a prototipagem evolutiva são conduzidos de maneira cíclica, de forma que a cada iteração do produto entregue, é feita uma avaliação com as partes interessadas e o *feedback* coletado dessa avaliação é usado para melhoria do produto para o próximo ciclo. Este método cíclico de desenvolvimento é diferente dos modelos tradicionais da Engenharia de Software, que conduzem as atividades de forma linear. Os métodos citados foram escolhidos para este projeto pois possibilitam maior adesão das partes interessadas, bem como *feedbacks* durante o desenvolvimento do software. Detalhes sobre os métodos são descritos nas Seções 2.1.1 e 2.2.1.

2.1 Etapa de Design

Nesta etapa é preciso coletar requisitos do produto e entender as demandas do público alvo. Uma abordagem de coprodução minimiza possibilidades de não adesão ao produto. Neste sentido, este projeto empregou o Design Participativo para aproximar usuários alvo do processo de construção da solução.

2.1.1 *Design Participativo*

O Design Participativo é uma metodologia de design que inclui os usuários finais do sistema nas etapas de concepção e avaliação. Essa participação mais próxima do usuário final melhora a eficiência e a qualidade do sistema e de sua criação (MULLER; HASLWANTER; DAYTON, 1997). Ao juntar o design participativo com a prototipagem evolutiva, é possível encontrar problemas de design antes que muito investimento seja feito no produto final.

Entre as práticas aplicadas no DP, cita-se a condução de oficinas com representantes dos usuários alvo para discussão, prototipagem e avaliação conjunta.

2.2 Etapa de Desenvolvimento

Nesta etapa são conduzidas as atividades de implementação e teste do produto. Neste projeto foram adotadas as metodologias de prototipagem evolutiva e Scrum, de modo que o produto foi desenvolvido de forma iterativa, com entregas parciais dos requisitos.

Como boas práticas de desenvolvimento, a programação em pares e a revisão de código foram adotadas.

2.2.1 Prototipagem Evolutiva

A prototipagem evolutiva acontece de forma gradual e cíclica. Protótipos evolutivos ou evolucionários (*evolutionary*) são criados nas fases iniciais do projeto e refinados no decorrer do processo de desenvolvimento do software, podendo ser interpretados como liberações. Incrementos de funcionalidade são incorporados ao protótipo, que, tendo sua fidelidade gradualmente aumentada, se torna o software final (HEKMATPOUR, 1987).

2.2.2 Scrum

Durante o desenvolvimento, o time envolvido no projeto da RUFUS utilizou uma adequação da metodologia *Scrum*. O time é dividido em 3 equipes: equipe de desenvolvimento web, equipe de design e equipe de desenvolvimento móvel. Os *sprints* acontecem semanalmente. O time se reúne para mostrar e avaliar o que foi desenvolvido na última semana e cada equipe se reúne para dividir as tarefas e discutir detalhes mais técnicos.

2.2.3 Programação em Pares

A técnica de programação em par faz parte das técnicas de desenvolvimento ágil. Ela consiste em dois programadores dividindo a mesma máquina, de forma que o *piloto* escreve o código, enquanto o *navegador* analisa o código que está sendo escrito, ajudando a corrigir possíveis falhas técnicas e de design de código. É comum os programadores trocarem de papel com frequência durante a realização dessa técnica. O uso dela costuma produzir códigos mais simples de entender e ler, além de contribuir para o aprendizado de ambos os envolvidos, independentemente do nível de conhecimento (COCKBURN; WILLIAMS, 2000). Durante o desenvolvimento da plataforma RUFUS, a equipe de desenvolvimento da aplicação móvel adotou o uso dessa técnica, principalmente para integrar os novos programadores à base de código, uma vez os membros da equipe podem mudar com frequência.

Para problemas mais complexos, a equipe utilizou-se da técnica de *Mob Programming*, uma técnica semelhante a programação em par, em que mais pessoas atuam como *navegador* durante a execução.

É importante destacar que o uso dessas técnicas pode tornar o desenvolvimento mais lento no curto prazo, pois elas alocam dois ou mais programadores para resolverem o mesmo problema. Porém, a longo prazo, a qualidade do código compensa o tempo extra.

2.2.4 Revisão de Código

Para manter a clareza, coerência e qualidade do código como um todo, foi adotada a técnica de revisão de código. O autor dessa monografia, o qual é responsável por coordenar a equipe de desenvolvimento da aplicação móvel da RUFUS, é responsável por revisar as mudanças de código entregues sempre que uma nova funcionalidade é completada. Ao terminar a revisão, ele pode aceitar as mudanças do código ou pedir alterações, repetindo o ciclo de revisão.

2.2.5 Ferramentas Utilizadas

Para desenvolver o primeiro protótipo da Narrativa Invertida, a designer utilizou a ferramenta de design de interface Adobe XD ¹. Em seguida, ao iniciar a implementação do protótipo móvel, a equipe utilizou o motor de jogos *Unity Game Engine* ², a linguagem de programação C#, que utiliza o paradigma de Programação Orientada a Objeto (POO) e o Git ³ para o gerenciamento e controle de versões do projeto. Por fim, para o planejamento e documentação foram utilizados o Trello ⁴ e o *draw.io* ⁵.

2.3 Etapa de Avaliação

Avaliações podem ser realizadas desde o início do projeto, com um protótipo em baixa fidelidade, por exemplo. Essa etapa é importante para garantir que todas as partes interessadas no projeto estejam alinhadas e evitar que ajustes robustos sejam realizados apenas ao fim do projeto.

Neste trabalho foram conduzidos testes de usabilidade (LEWIS, 2006), com as seguintes etapas: questionário demográfico inicial, seguido por uma interação no produto seguindo um roteiro, aplicação dos questionários *System Usability Scale (SUS)* (BROOKE, 2013) e *Self-Assessment Manikin (SAM)* (HAYASHI *et al.*, 2016; BRADLEY; LANG, 1994) e, por fim, uma entrevista semiestruturada. Para a aplicação do SUS e do SAM, foi utilizada a plataforma *EmoFrame*⁶, desenvolvida por Santos (2022).

¹ <https://www.adobe.com/br/products/xd.html>

² <https://unity.com/>

³ <https://git-scm.com/>

⁴ <https://trello.com/>

⁵ <https://app.diagrams.net/>

⁶ <https://emoframe.icmc.usp.br/>

2.3.1 System Usability Scale (SUS)

O SUS é uma ferramenta usada para avaliar a usabilidade do sistema. Ela consiste em um questionário com 29 perguntas, avaliadas de 1 a 5, sendo 1 concordo fortemente e 5 discordo fortemente. Cada resposta recebe uma pontuação. Essa pontuação é usada para o cálculo do *score*, o qual varia de 0% a 100%. A análise da usabilidade do sistema é determinada pelo *score* da seguinte forma:

- $score < 60$ = Inaceitável
- $60 \leq score < 70$ = Ok
- $70 \leq score < 80$ = Bom
- $80 \leq score \leq 90$ = Excelente
- $score > 90$ = Melhor usabilidade possível

2.3.1.1 Heurísticas de Nielsen

Para melhor analisar as respostas das perguntas do SUS, foram usadas as heurísticas de usabilidade propostas por Nielsen (1994). Essas heurísticas são dez princípios de design de interação que corroboram para melhoria da usabilidade de um sistema. As heurísticas usadas neste trabalho estão enumeradas abaixo:

1. Correspondência entre o sistema e o mundo real;
2. Controle e liberdade para o usuário;
3. Consistência e padronização;
4. Reconhecimento ao invés de memorização;
5. Flexibilidade e eficiência;
6. Estética e design minimalista;
7. Ajuda aos usuários a reconhecer, diagnosticar e se recuperar de erros.

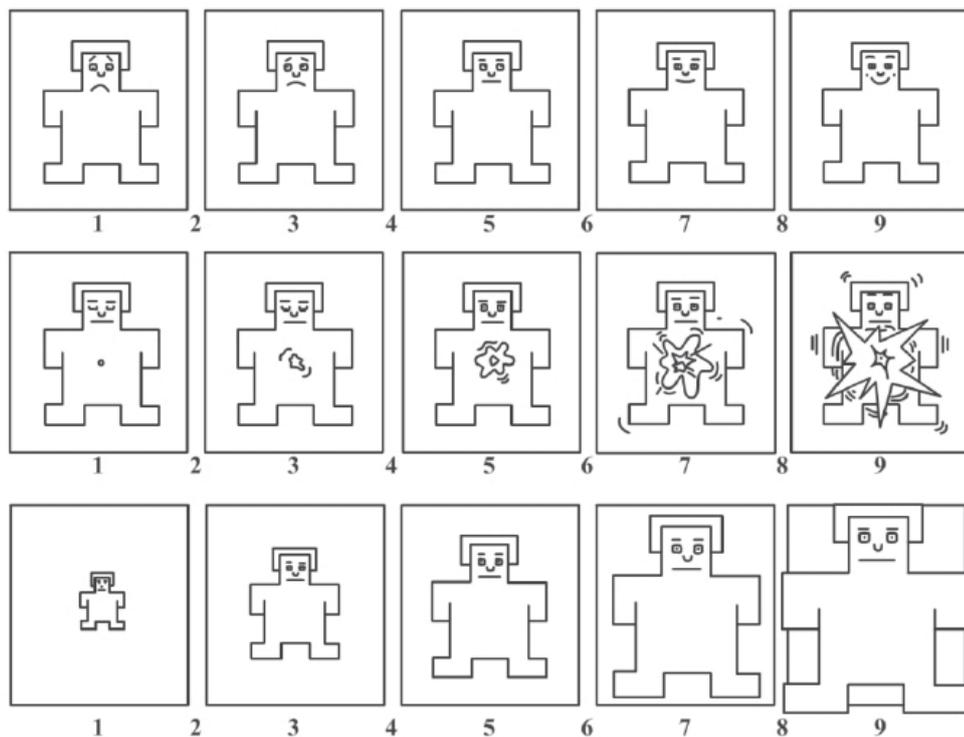
As heurísticas acima descritas servem como categorização das questões da ferramenta SUS, facilitando a análise dos resultados de tal instrumento. As outras três finais não apoiam diretamente as questões definidas para o questionário SUS.

2.3.2 Self-Assessment Manikin (SAM)

O SAM é um instrumento pictográfico, composto por três domínios: Satisfação, Motivação e Sentimento de controle do usuário. Como resposta, o usuário escolhe um valor correspondente à uma escala de 1 a 9, sendo 1 a 4 respostas negativas, 5 neutra e 6 a 9 respostas positivas (BRADLEY; LANG, 1994; HAYASHI *et al.*, 2016).

A Figura 1 ilustra o instrumento original.

Figura 1 – SAM (BRADLEY; LANG, 1994).



DESENVOLVIMENTO DA NARRATIVA INVERTIDA - APLICAÇÃO MÓVEL

As mecânicas disponibilizadas inicialmente na plataforma RUFUS (*quiz*, encaixe, coleta e narrativa) não davam a liberdade ao jogador (população de interesse dos profissionais da saúde) de instanciar sua própria solução computacional e se expressar nessa criação, um dos preceitos da EUD.

O contexto dos trabalhos de [Verhalen et al. \(2022\)](#), porém, trouxe a demanda pela oferta de jogos na RUFUS que possam permitir o diálogo sobre temas sensíveis como a morte. Esse diálogo deve ser mediado pelo profissional de saúde. Neste sentido, a mecânica de Narrativa Invertida foi idealizada, a qual consiste em um jogo em que é possível criar uma história sequencial, representada por páginas, usando elementos ou *assets* configurados pelo profissional da saúde. O ato de criar uma narrativa pode exercitar a liberdade, criatividade e expressão do jogador.

3.1 Atividades Realizadas

As atividades foram conduzidas em três etapas, conforme previamente descrito: design, desenvolvimento e avaliação. Essas etapas ocorreram de forma cíclica, como discorrido na Seção 2.3. A seguir, a condução das etapas, os artefatos gerados nas mesmas e os resultados são descritos.

3.1.1 Etapa de Design

A primeira etapa, de design, iniciou-se com a criação de um protótipo de média fidelidade do jogo de Narrativa Invertida. Esse protótipo foi criado pela designer do time e avaliado pelo responsável por este projeto para entender os requisitos expostos. As Figuras 2, 3 e 4 ilustram algumas telas do protótipo.

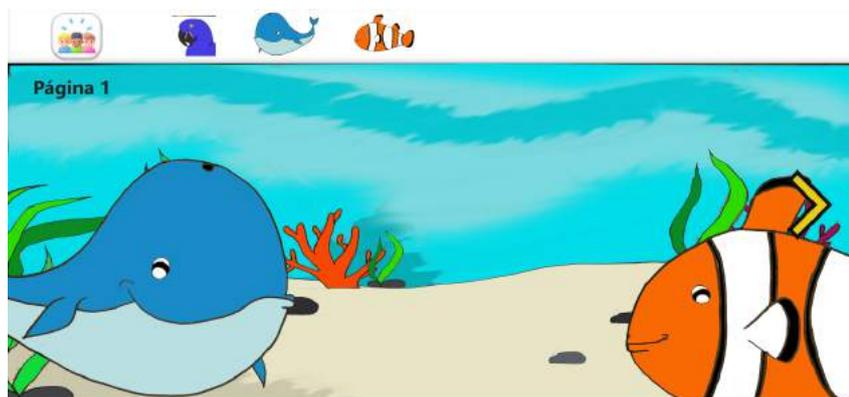
Em seguida, foi realizada uma entrevista semiestruturada com profissionais da saúde para validar o protótipo. Após essa avaliação, a designer se reuniu com as equipes de desenvolvimento da RUFUS para realizar um *brainstorming*. Nesse *brainstorming*, foi realizada a etapa de coleta de requisitos da mecânica (a partir do protótipo) e iniciou-se, também, a documentação para o desenvolvimento.

Figura 2 – Protótipo da Designer - Criação de Personagem.



Fonte: arquivos da designer proponente da mecânica.

Figura 3 – Exemplo de tela do protótipo - Personagens.



Fonte: arquivos da designer proponente da mecânica.

Figura 4 – Exemplo de tela do protótipo - Balões.



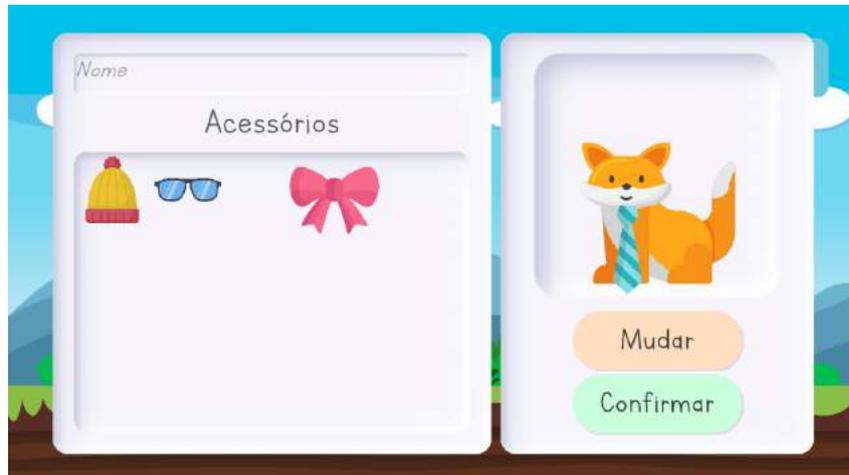
Fonte: arquivos da designer proponente da mecânica.

3.1.1.1 Fluxo de Execução da Narrativa Invertida

A mecânica de Narrativa Invertida possui 4 etapas durante a interação do jogador. A primeira etapa, consiste em uma breve explicação da mecânica para o jogador. A segunda etapa, permite ao jogador configurar personagens para usar em sua história. Essa criação se dá usando

as imagens de personagens e acessórios configuradas para o jogo pelo profissional de saúde na interface Web da RUFUS, permitindo ao jogador escolher um personagem e adicionar acessórios nele para customizá-lo. Essa etapa de escolha de personagem se assemelha àquela já disponível na mecânica de Narrativa (*storytelling*), ilustrada na Figura 5.

Figura 5 – Criação de Personagem na Mecânica de Narrativa.



Fonte: Elaborada pelo autor.

Os personagens criados poderão ser usados na terceira etapa, a qual inicia a criação da história de fato. Nessa etapa, o jogador pode navegar entre páginas, criar novas páginas, ordenar as páginas e adicionar elementos à página selecionada, sendo estes elementos os seguintes: personagens, incluindo os que foram criados por ele, objetos, balões de fala, efeito sonoro e imagem de fundo. Também é possível remover esses elementos da página. Por fim, o jogador pode ir para a quarta etapa, que consiste em visualizar a história criada, podendo passar entre as páginas sem a obstrução da mesma por elementos de interface. Para ordenar as páginas, foi pensado o menu de visão de páginas, demonstrado na Figura 6, o qual ilustra uma versão por cima de todas as páginas do jogo, permitindo arrastá-las e ordená-las entre si.

Figura 6 – Menu de Visualização de Páginas.



Fonte: arquivos da designer proponente da mecânica.

3.1.2 Etapa de Desenvolvimento

A segunda etapa, de desenvolvimento, aconteceu de forma iterativa, conforme a metodologia de Prototipagem Evolutiva da Engenharia de Software (HEKMATPOUR, 1987; PRESSMAN, 2011).

A equipe de desenvolvimento da aplicação móvel, coordenada pelo autor desta monografia, utilizou o motor de jogos Unity ¹ para desenvolver um protótipo para dispositivo móvel baseado nos requisitos coletados na etapa anterior. Toda semana, a equipe de desenvolvimento da aplicação móvel se reunia para dividir as tarefas entre si e avaliar as que estavam concluídas. Também foram realizadas reuniões semanais com o time geral, nas quais era apresentado o estágio do protótipo naquela semana e era feito um *brainstorming* coletando *feedback* de todos do time.

Como mencionado nas Seções 2.2.4 e 2.2.3, também foram utilizadas as técnicas de revisão de código e programação em pares e em *mob* durante o desenvolvimento da aplicação móvel.

3.1.2.1 Implementação da Aplicação Móvel

Devido ao escopo do projeto e tempo para produção deste texto, não foi possível implementar no dispositivo móvel todas as etapas descritas na Seção 3.1.1.1. Porém, a principal etapa, de criação da história, foi implementada quase completamente, faltando apenas a adição de elementos sonoros e a ordenação de páginas.

Seguindo o protótipo inicial entregue pela designer, a tela inicial do aplicativo consiste em uma página branca, com botões ao topo para acessar os diferentes elementos disponíveis e botões laterais para troca de página, como também um botão para criação de uma nova página. Na concepção inicial, o botão direito de troca de página também seria responsável pela criação de uma nova página. Entretanto, durante a implementação, observou-se que ficava confuso essa interação para o jogador, então, foi adicionado um botão específico para a criação de página.

Para adicionar um elemento, é preciso acessar o menu específico do mesmo. Por exemplo, para trocar a imagem de fundo, é preciso acessar o menu de fundo (*vide* Figuras 7 (protótipo) e 8 (implementação do protótipo)).

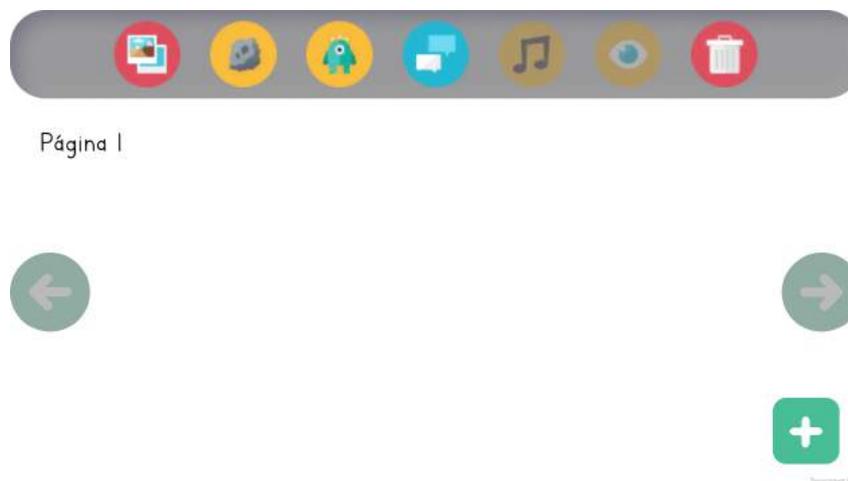
¹ <https://unity.com/>

Figura 7 – Tela inicial - Protótipo de média fidelidade da Designer



Fonte: Arquivos da designer proponente da mecânica.

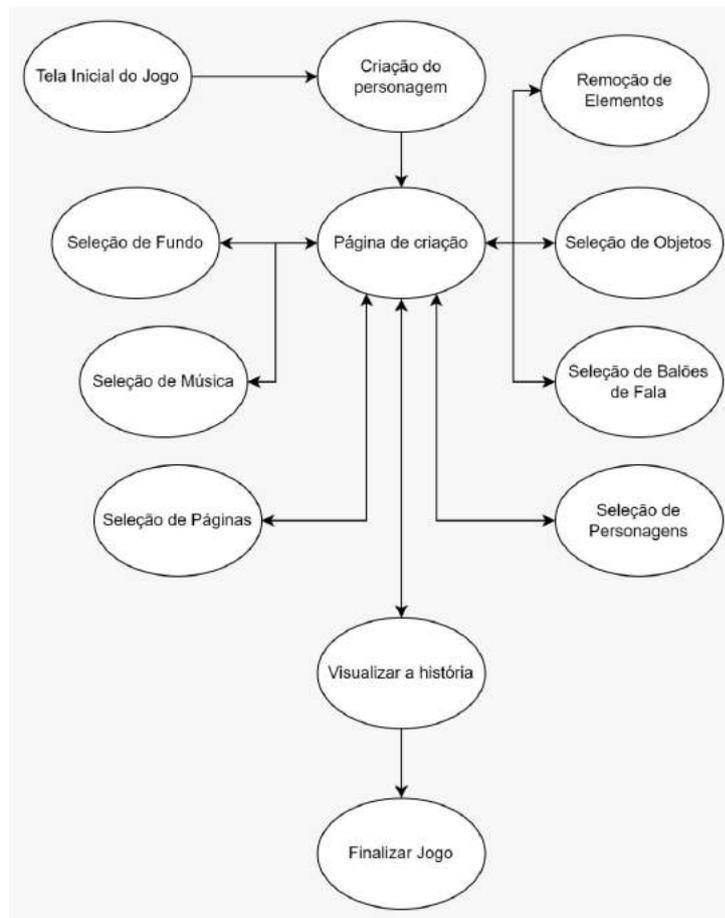
Figura 8 – Tela inicial - Protótipo da aplicação Móvel.



Fonte: Elaborada pelo autor.

Os passos para criação de uma história na Narrativa Invertida são os seguintes: 1) troca-se o fundo da primeira página, 2) adiciona-se os personagens, posicionando-os, mudando seu tamanho e rotação, 3) adiciona-se os objetos e balões de fala com texto, 4) cria-se uma nova página e adiciona-se os elementos na mesma para dar continuidade à história. A Figura 9 ilustra o fluxo de criação de uma história seguindo esses passos. Ressalta-se que esses não são passos lineares e podem ocorrer em ordens diferentes.

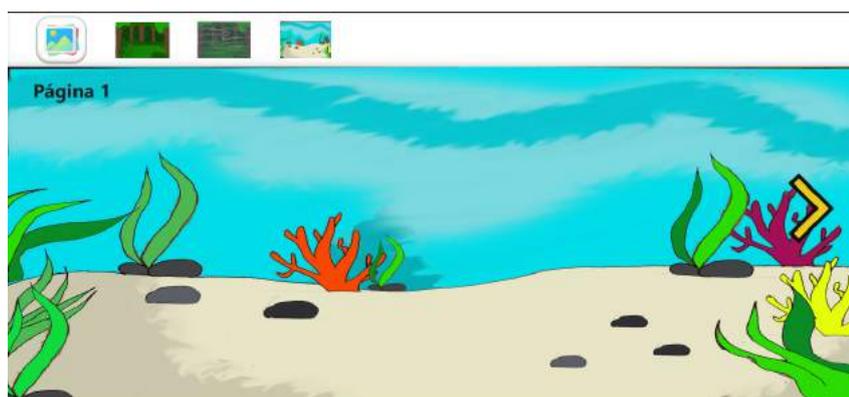
Figura 9 – Fluxo de criação de uma narrativa.



Fonte: Elaborada pelo autor.

As Figuras 10 e 11 representam o passo 1 (escolha da imagem de fundo da primeira tela), no protótipo e na versão implementada, respectivamente.

Figura 10 – Seleção de Fundo - Protótipo de média fidelidade da designer.



Fonte: Arquivos da designer proponente da mecânica.

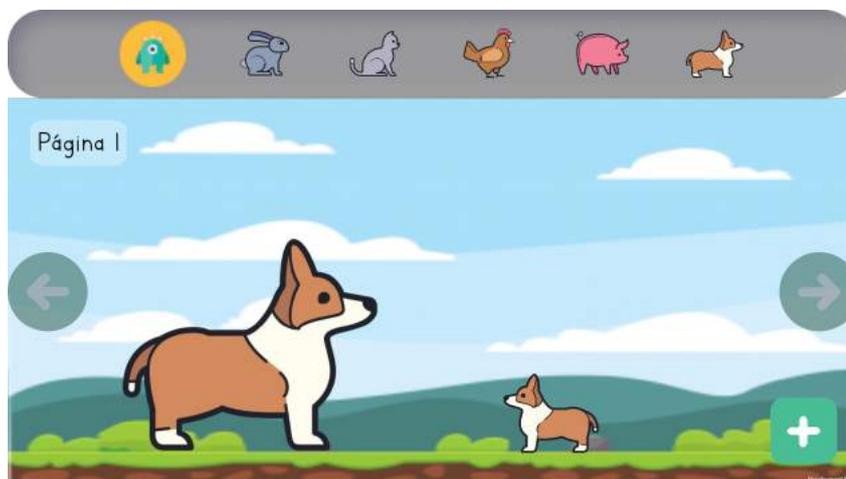
Figura 13 – Arrastando Personagem para a Página.



Fonte: Elaborada pelo autor.

Os personagens podem ter diferentes posições, tamanhos e orientações, conforme o ilustrado na Figura 14.

Figura 14 – Personagens com Diferentes Posições, Tamanhos e Orientações.



Fonte: Elaborada pelo autor.

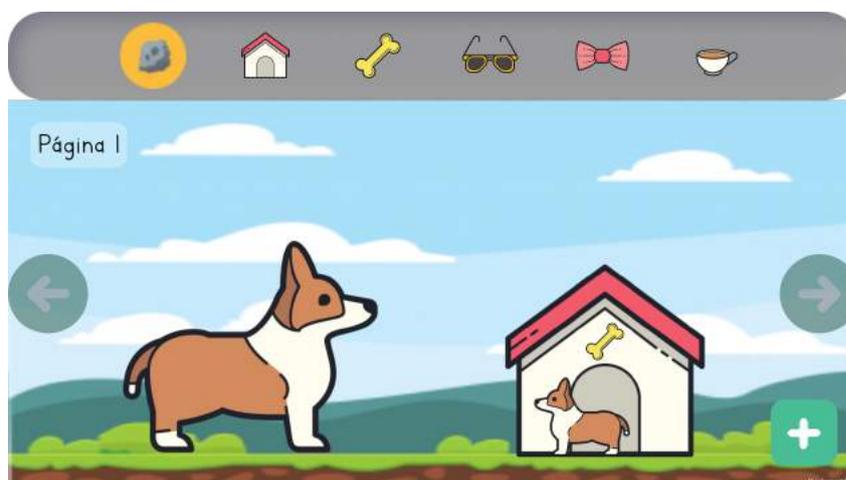
A Figura 15 ilustra o passo 3 (adicionar objetos e balões de fala), no protótipo idealizado pela designer. As Figuras 16, 17 e 18, por sua vez, ilustram este passo na implementação da mecânica (adicionar objetos na cena, adicionar balões e adicionar balões com texto, respectivamente).

Figura 15 – Adicionando Balão de Fala - Protótipo de média fidelidade elaborado pela designer.



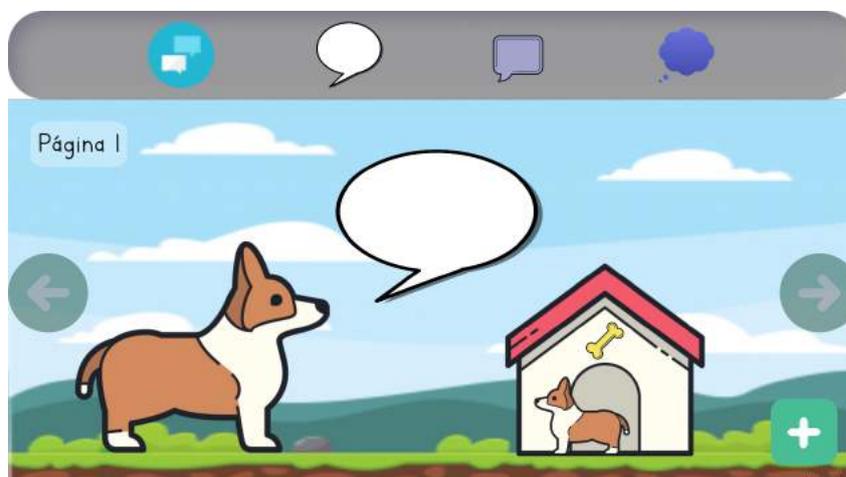
Fonte: Arquivos da designer proponente da mecânica.

Figura 16 – Adicionando Objetos - Protótipo da aplicação móvel



Fonte: Elaborada pelo autor.

Figura 17 – Adicionando Balão de Fala - Protótipo da aplicação móvel.



Fonte: Elaborada pelo autor.

Figura 18 – Balões de Fala com Texto.



Fonte: Elaborada pelo autor.

O passo 4 é uma replicação dos passos anteriores para a criação de novas cenas/páginas. A Figura 19 ilustra uma nova página montada na versão implementada da mecânica.

Figura 19 – Nova Página Montada.



Fonte: Elaborada pelo autor.

Outras mudanças entre a aplicação móvel e o protótipo elaborado pela designer incluem: a interação de rotação dos elementos acontece com um duplo toque rápido sobre o elemento no dispositivo móvel, enquanto a ideia inicial seria por meio de um menu de rotação que apareceria sobre o elemento selecionado; o menu de remoção, demonstrado na Figura 20, não mostrava qual elemento estava selecionado para remoção na página em si, mas somente no menu lateral, o que podia causar confusão caso houvesse múltiplos elementos com a mesma imagem. Portanto, foi adicionada uma indicação sobre o elemento selecionado para remoção na página também, conforme ilustração na Figura 21.

Figura 20 – Menu de Remoção - Protótipo de média fidelidade elaborado pela designer.



Fonte: Arquivos da designer proponente da mecânica.

Figura 21 – Menu de Remoção - Protótipo da aplicação móvel.



Fonte: Elaborada pelo autor.

3.1.3 Etapa de Avaliação

A terceira e última etapa, de avaliação, aconteceu de forma iterativa também. Como descrito na Seção 3.1.1. Após o desenvolvimento do primeiro protótipo, foi realizada uma avaliação com profissionais da saúde mental. Eram 2 pessoas, da área de psicologia, uma professora em atuação e a outra aluna de graduação, ambas com experiência no tema de luto infantil. Elas foram entrevistadas de forma remota e puderam interagir com o protótipo por meio de uma simulação do mesmo em formato de apresentação de slides, uma vez que o Adobe XD² é uma aplicação paga e, portanto, inacessível de forma remota.

A recepção das profissionais de saúde foi positiva e foi sugerido que adicionasse a funcionalidade de poder reproduzir um áudio lendo as falas escritas pelo jogador. Essa funcionalidade apoiaria questões de acessibilidade, porém, ainda não foi implementada, devido a sua complexidade.

² <https://www.adobe.com/br/products/xd.html>

Após finalizar a primeira versão do protótipo interativo e de alta fidelidade da aplicação móvel nesta mecânica, um teste de usabilidade (LEWIS, 2006) foi conduzido pelo autor desta monografia, com a designer proponente da mecânica de narrativa invertida e com sete usuários voluntários diferentes. Os sete voluntários foram escolhidos por conveniência e são alunos da graduação em Computação do ICMC/USP.

Inicialmente os voluntários responderam ao questionário para levantar o perfil demográfico, em seguida interagiram com o protótipo por meio de um roteiro pré-definido, responderam aos questionários SUS e SAM e, por fim, participaram de uma entrevista semiestruturada.

É importante ressaltar que, a fim de manter o anonimato dos usuários voluntários dos testes de usabilidade, esses serão referenciados nesta monografia como U1, U2, e assim em diante.

3.1.3.1 Testes de Usabilidade

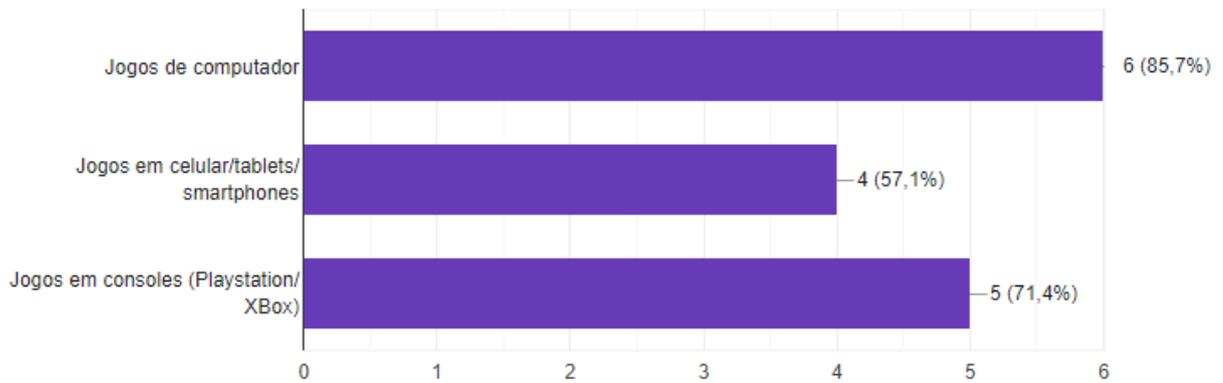
Os testes de usabilidade foram conduzidos em 5 momentos: Primeiro, o usuário preencheu ao questionário demográfico, a fim de obter dados sobre os perfis. Também assinaram o termo de consentimento livre e esclarecido. O segundo momento correspondeu à interação com o protótipo e roteiro, descrita na Seção 3.1.3.2. Após a interação, foram aplicados os questionários SUS e SAM, sendo seus resultados descritos nas Seções 3.1.3.3 e 3.1.3.4, respectivamente. O quarto momento consistiu na entrevista semiestruturada, descrita na Seção 3.1.3.5. E o quinto e último momento, consistiu nos agradecimentos e encerramento da avaliação.

Os testes foram aplicados com sete usuários voluntários, cujos perfis estão descritos no Quadro 1. Desses sete usuários voluntários, um reside em Campinas, enquanto os outros residem em São Carlos e todos se identificam com o sexo masculino. Também, todos usuários possuem alguma experiência com jogos digitais. Os tipos de jogos que os usuários estão habituados a jogar está demonstrado na Figura 22.

Quadro 1 – Perfil de usuários voluntários.

Usuário	Idade	Formação Acadêmica	Atuação Profissional
U1	22	Cursando Bach. em Ciên. de Comp.	Estagiário em desenv. de soft.
U2	23	Bach. em Ciên. de Comp.	Engenheiro de Software
U3	31	Tecnólogo em Análise e Desenv. de Sist.	Engenheiro de Software
U4	25	Cursando Bach. em Ciên. de Comp.	Analista de Seg. da Info.
U5	22	Cursando Engenharia de Produção	Estudante
U6	22	Cursando Bach. em Ciên. de Comp.	Estudante
U7	24	Cursando Bach. em Ciên. de Comp.	Desenvolvedor <i>Mobile</i>

Figura 22 – Tipos de jogos que os voluntários estão habituados a jogar.



3.1.3.2 Interação com Roteiro

A interação com roteiro aconteceu no 2º momento do teste de usabilidade, como mencionado na Seção 3.1.3.1. Nessa interação, foi solicitado aos voluntários que realizassem as tarefas descritas no Quadro 2 sem ajuda, *a priori*. Porém, caso o voluntário demorasse de um minuto a um minuto e meio, era possível intervir e ajudá-lo a realizar a tarefa. Esse tipo de intervenção aconteceu durante a tarefa de remoção da imagem de fundo com três dos sete usuários e com a tarefa de mudar o texto do balão, com dois deles. Durante a interação, os usuários foram observados pelo avaliador, o qual também tomou notas das ações e dificuldades percebidas. Essas anotações são refletidas nas Seções 3.1.3.7 e 3.1.3.6.

Quadro 2 – Tarefas da Interação com Roteiro

Tarefa
Mude a imagem de fundo
Adicione um personagem à página
Interaja com o personagem adicionado, movendo, trocando de direção e tamanho
Adicione um objeto à página
Posicione o objeto sobre o personagem
Posicione o personagem sobre o objeto
Adicione um balão de fala à página
Posicione o balão
Escreva algo no balão de fala
Troque o texto escrito no balão de fala
Crie uma nova página
Volte para a primeira página
Remova o fundo da primeira página
Selecione todos objetos da página para remoção
“Deselecione” todos objetos da página para remoção
Monte uma historinha de até 3 páginas

3.1.3.3 Resultados do System Usability Scale (SUS)

Durante o 3º momento, os usuários responderam ao questionário do SUS sobre o uso do aplicativo. A Tabela 1 ilustra o *score* e *status* de usabilidade de cada usuário. Para interpretar melhor as respostas, as perguntas do SUS foram divididas e categorizadas de acordo com as heurísticas de Nielsen (1994).

Tabela 1 – *Score* dos Usuários

Usuário	<i>Score</i>	Status de Usabilidade
U1	93,75	Melhor usabilidade possível
U2	86,61	Excelente
U3	66,07	Ok
U4	75,00	Bom
U5	95,54	Melhor usabilidade possível
U6	80,36	Excelente
U7	58,04	Inaceitável

Fonte: Dados da pesquisa.

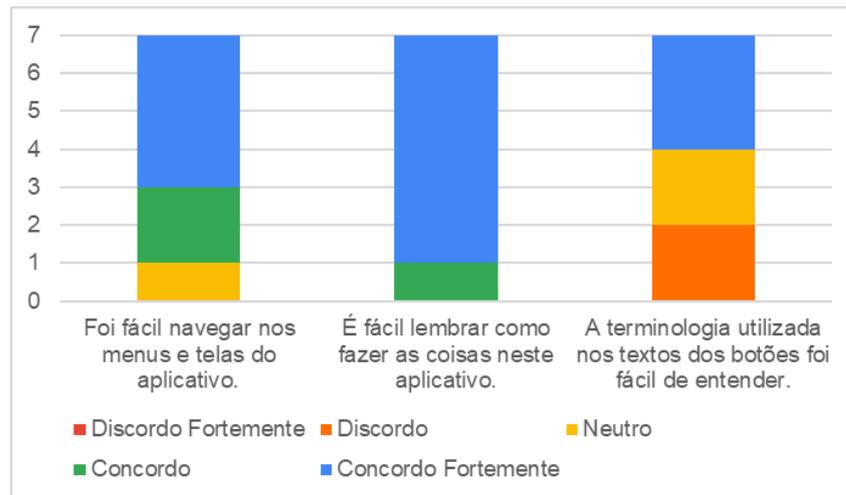
Correspondência entre o sistema e o mundo real

De acordo com a definição dessa heurística, o design deve usar palavras, frases e conceitos familiares ao usuário, ao invés de um jargão interno. Para isso, deve-se seguir convenções do mundo real, de forma que a informação apareça em uma ordem natural e lógica. As perguntas que se enquadram nessa heurística estão listadas abaixo:

- “Foi fácil navegar nos menus e telas do aplicativo.”
- “É fácil lembrar como fazer as coisas neste aplicativo.”
- “A terminologia utilizada nos textos dos botões foi fácil de entender.”

A Figura 23 ilustra as respostas dos usuários às questões dessa heurística. A maioria apresentou respostas positivas em relação a facilidade de navegação e de lembrar como fazer as coisas. Enquanto que, se tratando da terminologia utilizada nos textos dos botões, as respostas foram mais mistas, uma vez que os botões não possuem texto. Essa ausência de texto pode ser uma falha de design, porém a maioria dos usuários conseguiu navegar de forma tranquila pelo aplicativo ainda assim.

Figura 23 – Correspondência entre o sistema e o mundo real.



Fonte: Elaborada pelo autor.

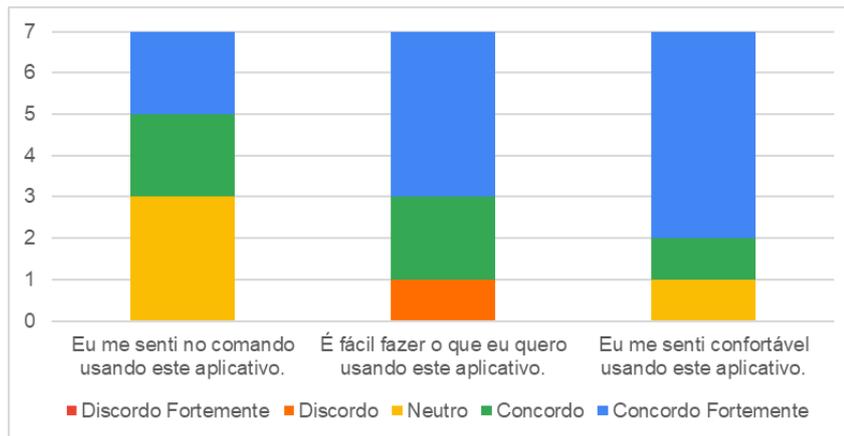
Controle e liberdade para o usuário

De acordo com a definição dessa heurística, usuários cometem erros com frequência. Portanto, é necessário que haja uma "saída de emergência" simples para que o usuário corrija seu erro e se sinta livre e sobre controle. As perguntas que se enquadram nessa heurística estão listadas abaixo:

- "Eu me senti no comando usando este aplicativo"
- "É fácil fazer o que eu quero usando este aplicativo"
- "Eu me senti confortável usando este aplicativo"

A Figura 24 ilustra as respostas dos usuários às questões dessa heurística. De acordo com as respostas, os usuários, em sua maioria, se sentiram confortáveis usando o aplicativo e disseram ser fácil fazer o que querem usando-o. Porém, metade dos voluntários se declarou neutro em relação a se sentir no comando. Um usuário não apontou ser fácil fazer o que quer no aplicativo. Os principais problemas que podem estar refletidos nessas respostas são as funções de remoção, inserir texto no balão de fala e mover um elemento sobre outro, as quais os usuários apresentaram maior dificuldade.

Figura 24 – Controle e liberdade para o usuário.



Fonte: Elaborada pelo autor.

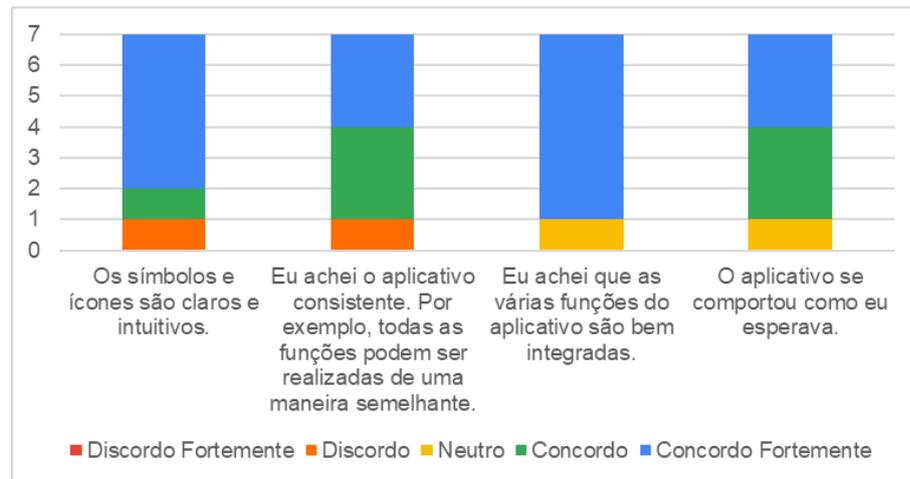
Consistência e padronização

De acordo com a definição dessa heurística, é importante manter uma consistência tanto interna quanto externa das ações, termos e situações que acontecem no aplicativo. Com essa consistência, os usuários precisam de um esforço menor para utilizar o aplicativo, pois mantém os padrões que eles estão acostumados. As perguntas que se enquadram nessa heurísticas estão listadas abaixo:

- "Os símbolos e ícones são claros e intuitivos"
- "Eu achei o aplicativo consistente..."
- "Eu achei que as várias funções do aplicativo são bem integradas"
- "O aplicativo se comportou como eu esperava"

A Figura 25 ilustra as respostas dos usuários às questões dessa heurística. Um dos sete voluntários ainda teve dificuldade em relação aos símbolos e ícones e discorda que o aplicativo seja consistente. Porém, os outros seis apresentaram respostas positivas sobre essas questões, significando que o aplicativo conseguiu manter a consistência e padronização interna e externa, e seguiu os padrões da indústria, em sua maior parte.

Figura 25 – Consistência e padronização.



Fonte: Elaborada pelo autor.

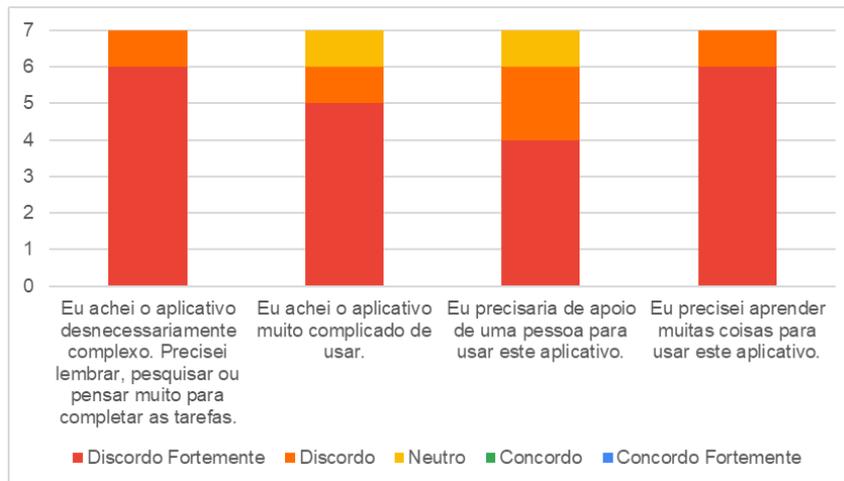
Reconhecimento ao invés de memorização

De acordo com a definição dessa heurística, ela busca reduzir a carga de memorização necessária ao usuário, mantendo os elementos com informações importantes sempre visíveis ou de fácil acesso. As perguntas que se enquadram nessa heurística estão listadas abaixo:

- "Foi fácil de aprender a usar este aplicativo."
- "A organização dos menus e comandos de ação (como botões e links) é lógica, permitindo encontrá-los facilmente na tela."
- "O aplicativo fornece todas as informações necessárias para completar as tarefas de forma clara e compreensível."
- "Eu achei o aplicativo muito complicado de usar."
- "Eu precisei aprender muitas coisas para usar este aplicativo."
- "Eu achei o aplicativo desnecessariamente complexo. Precisei lembrar, pesquisar ou pensar muito para completar as tarefas."
- "Eu precisaria de apoio de uma pessoa para usar este aplicativo."

A análise dessa heurística está separada entre questões negativas e positivas. A Figura 26 ilustra as respostas dos usuários às questões negativas dessa heurística. É possível notar que as respostas às perguntas negativas foram todas de discordância, e apenas duas neutras. Significando que o aplicativo é simples suficiente para que não haja a necessidade do usuário buscar ajuda ou aprender coisas novas para usá-lo.

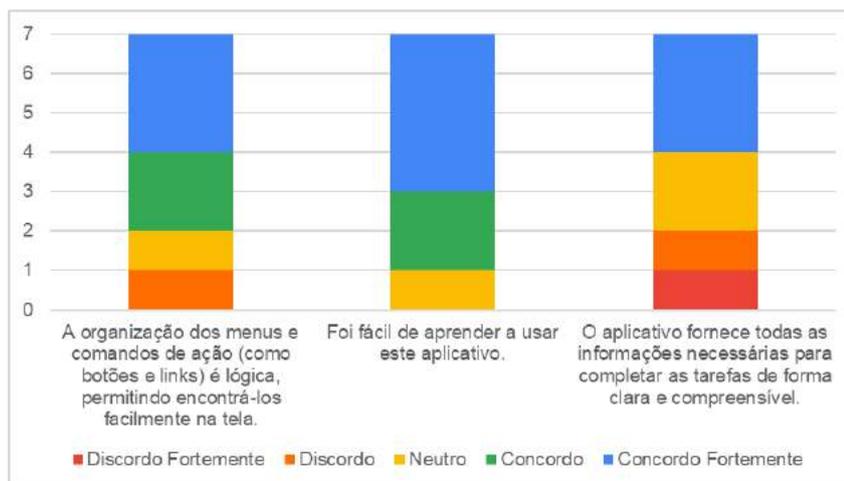
Figura 26 – Reconhecimento ao invés de memorização - questões negativas.



Fonte: Elaborada pelo autor.

A Figura 27 ilustra as respostas dos usuários às questões positivas dessa heurística. Aqui identificou-se uma falha de design, pois metade dos usuários sentiu que faltou informações necessárias para completar as tarefas de forma clara e compreensível. Entretanto, ainda assim, foi fácil aprender a usar o aplicativo e a organização dos menus e comandos de ação são lógicas para cinco dos sete voluntários. Esses apontaram maior dificuldade para entender as ações que precisavam segurar o dedo, como a de inserção de texto e a de remoção.

Figura 27 – Reconhecimento ao invés de memorização - questões positivas.



Fonte: Elaborada pelo autor.

Flexibilidade e eficiência

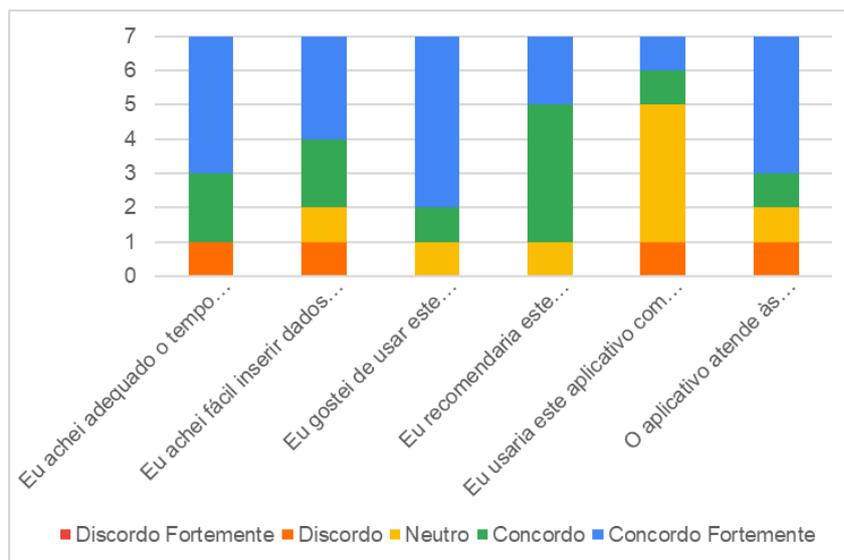
De acordo com a definição dessa heurística, o uso de atalhos, escondidos dos usuários inexperientes, pode acelerar a interação dos usuários mais experientes sem prejudicar a

experiência dos menos experientes. Permitir que os usuários configurem ações próprias e frequentes também favorece a eficiência no uso do aplicativo. As perguntas que se enquadram nessa heurística estão listadas abaixo:

- "Eu achei fácil inserir dados neste aplicativo."
- "Eu achei adequado o tempo que levei para completar as tarefas."
- "O aplicativo atende às minhas necessidades."
- "Eu recomendaria este aplicativo para outras pessoas."
- "Eu usaria este aplicativo com frequência."
- "Eu gostei de usar este aplicativo."

A Figura 28 ilustra as respostas dos voluntários às questões dessa heurística. A maioria se sente neutra em relação a usar o aplicativo com frequência, o que faz sentido, pois trata-se de um jogo terapêutico e seu uso, portanto, é para fins específicos e guiados. Cinco dos sete voluntários avaliaram as outras questões de forma positiva, enquanto um voluntário não sentiu que o aplicativo é flexível e eficiente o suficiente. Algumas sugestões para aumento da eficiência foram apontadas na Seção 3.1.3.7 pelos usuários, são estas: atalhos para remoção de itens e a possibilidade de duplicar a página anterior.

Figura 28 – Flexibilidade e eficiência.



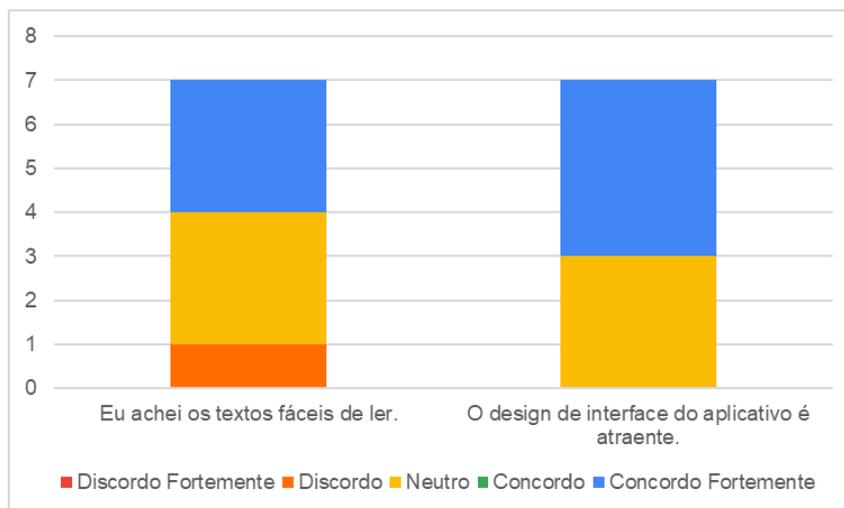
Fonte: Elaborada pelo autor.

De acordo com a definição dessa heurística, as interfaces devem dar destaque às informações mais relevantes e evitar informações adicionais, pois essas podem ocultar as mais relevantes. Também, deve-se evitar mostrar informações desnecessárias ou raramente usadas. As perguntas que se enquadram nessa heurística estão listadas abaixo:

- "O design de interface do aplicativo é atraente."
- "Eu achei os textos fáceis de ler."

A Figura 29 ilustra as respostas dos usuários às questões dessa heurística. O aplicativo não apresenta textos, além dos inseridos pelos usuários nos balões de fala, portanto, metade das respostas em relação à facilidade de ler os textos foram neutras. Ao mesmo tempo, três dos sete voluntários se sentiram neutros em relação ao design da interface, enquanto os outros quatro concordam fortemente que seu design é atraente.

Figura 29 – Estética e design minimalista.



Fonte: Elaborada pelo autor.

Ajude usuários a reconhecer, diagnosticar e se recuperar de erros

De acordo com a definição dessa heurística, as mensagens de erro devem ser expressas em linguagem clara (sem códigos de erro), indicando o problema de forma precisa e sugerindo soluções. As perguntas que se enquadram nessa heurística estão listadas abaixo:

- "Quando eu cometo um erro é fácil de corrigi-lo."
- "As mensagens de erro ajudam a corrigir os problemas."

Como o aplicativo não apresenta mensagens de erro, as respostas dos usuários foram consideradas neutras para essas questões. Essa é uma falha de design reconhecida e será tratada nas próximas iterações do protótipo.

3.1.3.4 Resultados do Self-Assessment Manikin (SAM)

O SAM avalia a experiência do usuário sobre três domínios: Satisfação, Motivação e Sentimento de Controle. As respostas dos voluntários estão ilustradas nas Figuras 30, 31 e 32 e a média e o desvio padrão na Tabela 2.

Figura 30 – Satisfação.

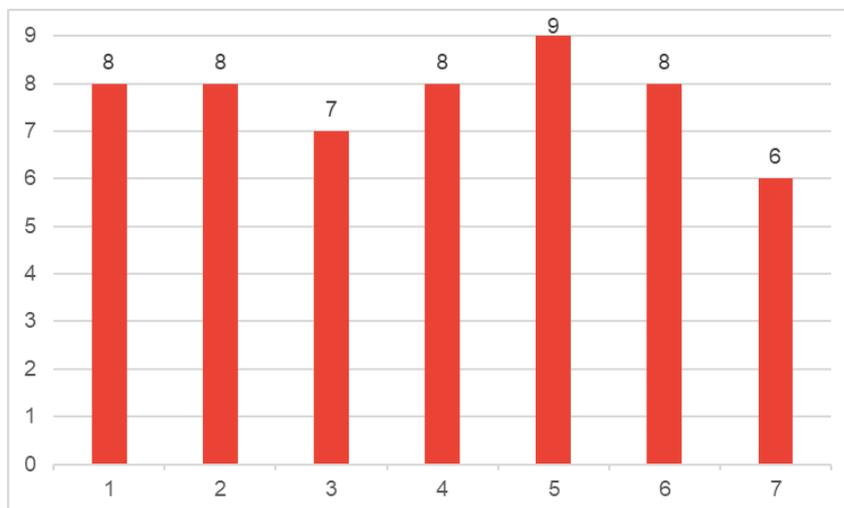


Figura 31 – Motivação.

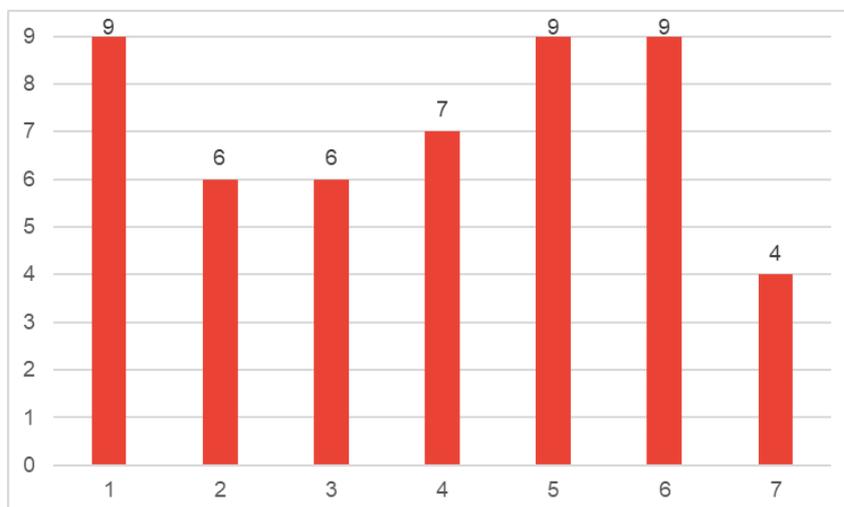
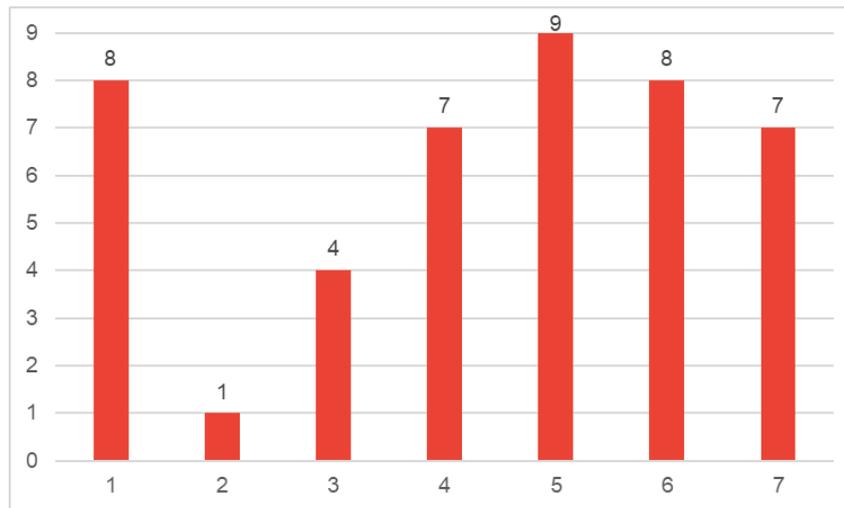


Figura 32 – Sentimento de Controle.



Fonte: Elaborada pelo autor.

Tabela 2 – Média e Desvio Padrão dos Domínios do SAM.

Domínio	Média	Desvio Padrão
Satisfação	8	0,881
Motivação	7	1,807
Sentimento de Controle	7	2,603

Fonte: Dados da pesquisa.

É possível observar que o domínio de **Satisfação** pode ser considerado positivo, uma vez que o desvio padrão é baixo e a média alta. O domínio de **Motivação**, por sua vez, apresenta resultados positivos, porém, o desvio padrão aumenta um pouco. Por fim, o desvio padrão do **Sentimento de Controle** se apresenta razoavelmente alto, o que indica uma maior prioridade para melhorias neste domínio.

3.1.3.5 Entrevista Semiestruturada

A entrevista semiestruturada foi composta das perguntas presentes no Quadro 3.

Quadro 3 – Perguntas Feitas Durante Entrevista.

Perguntas
O que achou do jogo no geral?
Considerando a escala de 1 a 5, sendo 1 muito fácil e 5 muito difícil, avalie as tarefas que foram propostas anteriormente.
O que você acha que pode ser aprimorado no jogo?
Use esse espaço para outros comentários que queira fazer.

Os comentários gerais dos entrevistados foram positivos, usando adjetivos como “simples”, “intuitivo” e “fácil” para descrever o aplicativo. Porém, cada um apontou os problemas

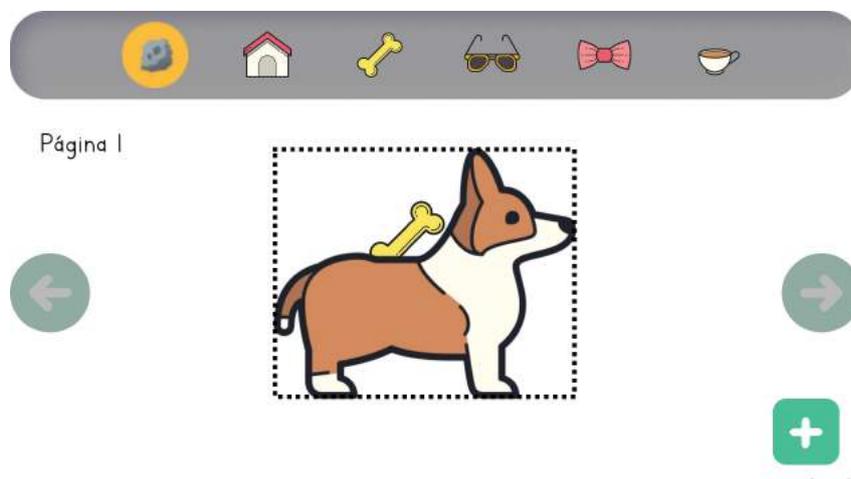
que tiveram durante a interação proposta. Esses problemas estão descritos na Seção 3.1.3.6.

Sobre a facilidade de realizar as tarefas propostas, a média de respostas foi de 1,5, sendo que todas as respostas foram entre 1 e 2, significando muito fáceis ou fáceis. Por fim, os aprimoramentos sugeridos pelos entrevistados estão descritos na Seção 3.1.3.7.

3.1.3.6 Problemas de Usabilidade Encontrados

Todos os voluntários apontaram e demonstraram dificuldades na remoção de elementos e na inserção de texto no balão de fala. É importante ressaltar que ambas as atividades envolvem a interação de segurar o dedo na tela por um curto período. Também vale destacar que, a primeira reação dos usuários ao tentar inserir um texto no balão de fala é de um toque curto sobre o balão, seguida de um toque duplo e, então, lembram que precisam segurar o dedo. Outra dificuldade apontada pelos usuários foi a ausência de instruções visuais ou textuais para a realização das tarefas. Por exemplo, o U4 comenta que, ao pedir para que ele remova o fundo, seu primeiro pensamento foi entrar no menu de fundo e selecionar um fundo vazio. O U7 comenta que não há indicações de como inserir o texto no balão de fala. Os voluntários também tiveram problema ao mudar a escala dos elementos, como de objetos e personagens, devido à implementação da interação de pinça, a qual necessita que um dedo se mantenha sobre o elemento sendo afetado e não impede que o outro dedo movimente outros elementos durante a interação, causando confusão ou movimentos indesejados. Por fim, o último problema identificado foi durante a movimentação dos elementos e, conseqüentemente, a atividade de colocar um elemento sobre o outro. O problema dessa interação é que o elemento pode ser selecionado em seu espaço transparente, causando confusão ao usuário. A Figura 33 ilustra, por exemplo, que o cachorro está a frente do osso, porém, se o usuário tentar movimentar o osso com o dedo, o cachorro que será movimentado, pois seu espaço transparente, presente dentro na borda pontilhada, está por cima do osso.

Figura 33 – Elementos Sobrepostos.



Fonte: Elaborada pelo autor.

3.1.3.7 Melhorias a serem realizadas

As melhorias propostas pelos voluntários incluem:

- Colocar contexto visual de como realizar as atividades de remoção e de inserção de texto;
- Usar *Tool tip* com os significados dos botões;
- Permitir duplicar a cena anterior ao criar uma nova, para dar continuidade à história;
- Poder visualizar a história criada como uma sequência de páginas, parecido com uma apresentação de slides;
- Poder visualizar a cena anterior por cima da atual, para fazer ajustes coerentes;
- Oferecer botão de deletar tudo e de cancelar a deleção;
- Permitir aumentar o tamanho da fonte dos balões de fala;
- Melhorar o movimento de pinça;
- Melhorar a seleção do personagem para movimentação.

A sugestão de visualizar a história de forma sequencial sem a interferência dos menus já é uma funcionalidade planejada, porém não foi possível implementá-la para o protótipo usado nessa avaliação.

CONCLUSÃO

4.1 Contribuições do Trabalho

A principal contribuição deste trabalho foi o design e desenvolvimento de uma mecânica nova para a plataforma RUFUS, a qual será disponibilizada e utilizada pelos profissionais de saúde quando concluída.

A nova mecânica, Narrativa Invertida, permite ao usuário se expressar de forma criativa, tornando-se uma importante ferramenta em tratamentos e estudos paliativos e sobre luto. Vale destacar a importância do processo de design em coprodução com o time da RUFUS e os profissionais de saúde, assim como as avaliações realizadas ao longo do projeto. Ao implementar uma primeira versão da aplicação móvel da RUFUS para essa mecânica, já foi possível coletar *feedbacks* importantes que identificaram falhas na usabilidade e implementação, os quais podem ser corrigidos em breve, aprimorando o sistema durante seu desenvolvimento.

4.2 Dificuldades e Limitações do Trabalho

A principal dificuldade encontrada durante o percurso deste trabalho foi a implementação das interações de toque, como múltiplo toque, toque prolongado e de movimento de pinça. Essas interações podem ser detectadas de múltiplas formas na Unity, usando o sistema de entrada novo ¹ ou o antigo ². Além disso, elas precisam ser comunicadas para os diferentes elementos do jogo e de interface, como também não podem ocorrer de forma simultânea. Um dos problemas encontrados com a interação de pinça, por exemplo, foi que ao fazer o movimento de pinça, esse não bloqueia outras interações como a de arrastar os elementos.

A solução adotada para comunicar as interações de toque aos objetos foi a implementação do sistema de mensagens da Unity ³, o qual permite separar o envio e a leitura da mensagem de evento. Assim, as interações de toque são capturadas e enviadas somente aos objetos interessados. Esses objetos interessados precisam somente implementar a interface correta para receber a mensagem desejada.

¹ <https://docs.unity3d.com/Manual/com.unity.inputsystem.html>

² <https://docs.unity3d.com/Manual/Input.html>

³ <https://docs.unity3d.com/Packages/com.unity.ugui@1.0/manual/MessagingSystem.html>

4.3 Considerações Sobre o Curso de Graduação

O curso de bacharelado em Ciências de Computação ajudou o autor deste trabalho a compreender os pequenos detalhes e complexidades do computador, desde uma perspectiva de hardware à uma de software. Esse conhecimento permite aprender novas tecnologias e conceitos com facilidade e entender como funcionam em um nível superficial sem buscar muito em profundidade.

Para esse trabalho, as disciplinas de SSC0770 - Introdução ao Desenvolvimento de Jogos Digitais, MAC0472 - Laboratório de Métodos Ágeis e MAC0413 - Tópicos Avançados em Programação Orientada a Objetos ajudaram a aprender a programar de forma mais eficaz, mantendo a qualidade do software conforme sua evolução. Já a disciplina de SCC0280 - Acessibilidade em Sistemas Computacionais fez com que fosse possível atentar mais aos requisitos de acessibilidade de sistemas computacionais e colocá-los como um importante fator de design ao desenvolver um novo sistema.

Além das disciplinas, as atividades de extensão, como o grupo de extensão Fellowship of the Game (FoG) e o projeto de iniciação científica da plataforma RUFUS, ajudaram este autor a evoluir como programador e como coordenador, pois me colocaram nessas posições com projetos reais e interessantes. Elas ajudaram, também a desenvolver habilidades sociais, como de comunicação.

Os pontos fracos encontrados no curso de bacharelado mencionado estão listados e descritos abaixo:

- SSC0124 - Análise e Projeto Orientados a Objetos: deveria focar mais na análise crítica de múltiplas soluções, invés de tentar encontrar a solução "perfeita" para um só problema. Outra melhoria seria o design e implementação de um sistema de software real durante a disciplina;
- 7600105 - Física Básica I: a primeira metade da disciplina é revisão do conteúdo do ensino médio e depois entra em conteúdos específicos de física que não são usados durante o restante do curso. Seria mais interessante focar na área de física computacional e fazer a revisão de conteúdo mais rapidamente;
- SSC0721 - Teste e Inspeção de Software: não cobre design movido a testes (TDD) e outros conteúdos extremamente relevantes para a área de teste e inspeção de software. Seria interessante atualizar a ementa da disciplina.

Também, é importante destacar as disciplinas listadas abaixo:

- SSC0770 - Introdução ao Desenvolvimento de Jogos Digitais: durante a disciplina são vistas todas as etapas de desenvolvimento de um jogo digital real e esse conhecimento é

aplicado durante o semestre em um projeto em grupo real, além de ter a mostra de jogos no fim do semestre, conhecida hoje como *Usp Game Link* (UGL);

- Disciplinas do primeiro e segundo semestre: são de extrema importância para aprender gerenciamento de memória e como a linguagem C funciona. Esse conhecimento serve de base para o aprendizado do resto do curso e de outras linguagens de programação;
- SSC0713 - Sistemas Evolutivos: o foco da disciplina é o desenvolvimento de um sistema evolutivo. Para isso, o professor atua como um orientador, guiando os alunos e ensinando os conteúdos necessários para construção do sistema.

Por fim, ficam as seguintes sugestões listadas abaixo:

- Projeto interdisciplinar e unificado entre as disciplinas do semestre;
- Mais disciplinas de Engenharia de Software como no IME da USP de São Paulo.

REFERÊNCIAS

BARBOSA, S.; SILVA, B. **Interação humano-computador**. [s.n.], 2010. Disponível em: <https://books.google.com/books?hl=en&lr=&id=qk0skwr_cekC&oi=fnd&pg=PP2&dq=Intera%C3%A7%C3%A3o+humano-computador+barbosa&ots=ROB85G11hu&sig=7Pt5GSNatkkxW900RL4YaUDRjDg>. Citado na página 19.

BRADLEY, M. M.; LANG, P. J. Measuring emotion: the self-assessment manikin and the semantic differential. **Journal of behavior therapy and experimental psychiatry**, Elsevier, v. 25, n. 1, p. 49–59, 1994. Citado 3 vezes nas páginas 7, 21 e 23.

BROOKE, J. Sus: a retrospective. **Journal of usability studies**, Usability Professionals' Association Bloomingdale, IL, v. 8, n. 2, p. 29–40, 2013. Citado na página 21.

CHEUNG, M. **Therapeutic games and guided imagery: tools for mental health and school professionals working with children, adolescents, and their families**. [S.l.]: Lyceum Books, 2006. Citado na página 17.

COCKBURN, A.; WILLIAMS, L. The costs and benefits of pair programming. 02 2000. Citado na página 20.

GRAMMENOS, D.; SAVIDIS, A.; STEPHANIDIS, C. Designing universally accessible games. **Comput. Entertain.**, v. 7, p. 8:1–8:29, 2009. Citado na página 17.

HAYASHI, E. C.; POSADA, J. E. G.; MAIKE, V. R.; BARANAUSKAS, M. C. C. Exploring new formats of the self-assessment manikin in the design with children. In: **Proceedings of the 15th Brazilian Symposium on Human Factors in Computing Systems**. [S.l.: s.n.], 2016. p. 1–10. Citado 2 vezes nas páginas 21 e 23.

HEKMATPOUR, S. Experience with evolutionary prototyping in a large software project. **SIGSOFT Softw. Eng. Notes**, Association for Computing Machinery, New York, NY, USA, v. 12, n. 1, p. 38–41, jan 1987. ISSN 0163-5948. Disponível em: <<https://doi.org/10.1145/24574.24577>>. Citado 2 vezes nas páginas 20 e 28.

LEWIS, J. R. Usability testing. **Handbook of human factors and ergonomics**, Wiley Online Library, v. 12, p. e30, 2006. Citado 2 vezes nas páginas 21 e 36.

LIEBERMAN, H.; PATERNÒ, F.; KLANN, M.; WULF, V. End-User Development: An Emerging Paradigm. **End User Development**, Springer, Dordrecht, p. 1–8, 10 2006. Disponível em: <https://link.springer.com/chapter/10.1007/1-4020-5386-X_1>. Citado na página 17.

MULLER, M. J.; HASLWANTER, J. H.; DAYTON, T. Chapter 11 - participatory practices in the software lifecycle. In: HELANDER, M. G.; LANDAUER, T. K.; PRABHU, P. V. (Ed.). **Handbook of Human-Computer Interaction (Second Edition)**. Second edition. Amsterdam: North-Holland, 1997. p. 255–297. ISBN 978-0-444-81862-1. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780444818621500777>>. Citado na página 19.

- NIELSEN, J. **10 usability heuristics for user interface design**. 1994. Disponível em: <<https://www.nngroup.com/articles/ten-usability-heuristics/>>. Citado 2 vezes nas páginas 22 e 38.
- PRESSMMAN, R. **Engenharia de software**. [S.l.]: Porto Alegre: AMGH, 2011. Citado 2 vezes nas páginas 19 e 28.
- RODRIGUES, K. R. d. H.; NERIS, V. P. d. A.; SOUZA, P. M.; ZAVARIZZ, R. G.; SILVA, J. W. da; SILVA, T. M.; VERHALEN, A. E. C. Rufus - uma plataforma de autoria para jogos digitais terapêuticos. In: **X Latin American Conference on Human Computer Interaction**. New York, NY, USA: Association for Computing Machinery, 2021. (CLIHC 2021). ISBN 9781450384896. Disponível em: <<https://doi.org/10.1145/3488392.3488407>>. Citado na página 17.
- SANTOS, S. S. dos. Emoframe: Um framework composto por avaliações de respostas emocionais de usuários frente a sistemas interativos. 03 2022. Citado na página 21.
- VERHALEN, A. E. C.; ZAVARIZZ, R. G.; SILVA, J. W. d.; SILVA, T. M.; NUNES, C. E. P.; DARIN, T. de G. R.; RODRIGUES, K. R. d. H. Telling your own story: Design and evaluation of a storytelling mechanic in a platform for serious games authoring. In: **Proceedings of the 21st Brazilian Symposium on Human Factors in Computing Systems**. New York, NY, USA: Association for Computing Machinery, 2022. (IHC '22). ISBN 9781450395069. Disponível em: <<https://doi.org/10.1145/3554364.3561610>>. Citado 2 vezes nas páginas 17 e 25.
- YUAN, B.; FOLMER, E.; HARRIS, F. C. Game accessibility: a survey. **Universal Access in the Information Society**, v. 10, n. 1, p. 81–100, mar. 2011. Citado na página 17.