

UNIVERSIDADE PAULISTA

**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

DAVID RODGERS NUNES G7364E-7

FERNANDO FERNANDES GOMES N3425F-0

JOÃO VITOR DA SILVA T915DJ-0

JUAN GABRIEL GENTIL DA CRUZ G74758-5

MATHEUS COSTA CAETANO PINTO R021ED-3

ROBERTA JAMILE DE OLIVEIRA G82BEA-2

PROJETO DE UM TOTEM DE MUSEU MULTITEMÁTICO

SOROCABA

2024

DAVID RODGERS NUNES G7364E-7
FERNANDO FERNANDES GOMES N3425F-0
JOÃO VITOR DA SILVA T915DJ-0
JUAN GABRIEL GENTIL DA CRUZ G74758-5
MATHEUS COSTA CAETANO PINTO R021ED-3
ROBERTA JAMILE DE OLIVEIRA G82BEA-2

PROJETO DE UM TOTEM DE MUSEU MULTITEMÁTICO

Trabalho de conclusão do semestre PIM (projeto integrado multidisciplinar) do curso de Análise e Desenvolvimento de Sistema apresentado à Universidade Paulista – UNIP.

Orientador: Prof. Esp. Reverdan Almeida
Sparinger

SOROCABA
2024

**DAVID RODGERS NUNES G7364E7
FERNANDO FERNANDES GOMES N3425F0
JOÃO VITOR DA SILVA T915DJ-0
JUAN GABRIEL GENTIL DA CRUZ G74758-5
MATHEUS COSTA CAETANO PINTO R021ED-3
ROBERTA JAMILE DE OLIVEIRA G82BEA-2**

PROJETO DE UM TOTEM DE MUSEU MULTITEMÁTICO

Trabalho de conclusão do semestre PIM (projeto integrado multidisciplinar) do curso de Análise e Desenvolvimento de Sistema apresentado à Universidade Paulista – UNIP.

Aprovado em:

BANCA EXAMINADORA

_____/_____/_____
Prof. Esp. Alex Sampaio
Universidade Paulista – UNIP

_____/_____/_____
Prof. Esp. Reverdan Almeida Sparinger
Universidade Paulista – UNIP

_____/_____/_____
Prof. Me. Waldir Antônio da Silva
Universidade Paulista – UNIP

SOROCABA
2024

RESUMO

Com o avanço da tecnologia, novos meios de comunicação e experiências sensoriais estão sendo criados, resultando em uma crescente integração da Tecnologia da Informação (TI) no cotidiano das pessoas, o que torna algo cultural na atualidade. Por isso instituições estão aderindo essa tendência. No presente trabalho foi desenvolvido o documento de um projeto para totens interativo que auxiliam na coleta de feedback, facilitando futuras melhorias. Utilizando de um software onde o cliente poderá avaliar a sua satisfação de acordo com o questionário predeterminado pelo proprietário do totem. Sendo coerente com os critérios de Engenharia de Software, foi definido os requisitos de negócios, requisitos funcionais não funcionais, normatizações, ciclo de vida e a implementação do protótipo dentro de uma instituição.

Palavras-Chave: Tecnologia, software, totem, projeto, instituições, documento, feedback, requisitos e protótipo.

ABSTRACT

With the advancement of technology, new means of communication and sensory experiences are being created, resulting in a growing integration of Information Technology (IT) in people's daily lives, which makes it something cultural today. That's why institutions are joining this trend. In the present work, a project document was developed for interactive totems that help in the collection of feedback, facilitating future improvements. Using a software where the customer can evaluate their satisfaction according to the questionnaire predetermined by the owner of the totem. Being consistent with the Software Engineering criteria, the business requirements, non-functional functional requirements, standardizations, life cycle and the implementation of the prototype within an institution were defined.

Keywords: Technology, software, totem, project, institutions, document, feedback, requirements and prototype.

LISTA DE ILUSTRAÇÕES

Figura 1 Ilustração planta baixa museu multitemático	14
Figura 2 Emoticons de satisfação.....	15
Figura 3 LGPD em um giro	19
Figura 4 Diagrama de classe	23
Figura 5 Diagrama de caso de uso.....	24
Figura 6 Diagrama de sequência.....	25
Figura 7 Fluxograma botão Mapa.....	26
Figura 8 Fluxograma botão História.....	27
Figura 9 Fluxograma botão Avaliação	28
Figura 10 Ciclo de vida do Software	38
Figura 11 Tela Principal	38
Figura 12 Tela Mapa	39
Figura 13 Tela História	39
Figura 14 Tela Avaliação.....	40
Figura 15 Tela Relatório.....	40
Figura 16 Tela Trello	42
Figura 17 Tela Discord	42

LISTA DE TABELAS

Tabela 1 Calculo CES 1.0.....	37
-------------------------------	----

LISTA DE QUADROS

Quadro 1 RF Tela Touch.....	30
Quadro 2 RF Questionário	31
Quadro 3 RF Teclado virtual	31
Quadro 4 RF Relatório	32
Quadro 5 RNF Linguagem C#	32
Quadro 6 RNF Windows Forms.....	33
Quadro 7 RNF Linguagem C#	33
Quadro 8 RN Utilização de ISO	35
Quadro 9 RN LGPD.....	36
Quadro 10 RN Home a lua.....	36

LISTA DE ABREVIATURAS E SIGLAS

CES – Customer Effort Score

GDPR – Regulamento Geral de Proteção de Dados

IDE – Ambiente de Desenvolvimento Integrado

IOT – Internet Of Things (Internet das coisas)

LGPD – Lei Geral de Proteção de Dados Pessoais

PIM – Projeto Integrado Multidisciplinar

POO – Programação Orientada a Objeto

QS – Qualidade de Software

RAM – Random Access Memory (Memória de acesso aleatório)

SFTP – Sistemas de Transferência de Arquivos Seguros

SSD – Solid-State Drive (Unidade de estado sólido)

TI – Tecnologia da Informação

SUMÁRIO

1	INTRODUÇÃO	11
2	OBJETIVOS	12
2.1	Objetivos Gerais.....	12
2.2	Objetivos específicos	12
2.3	Recursos	12
2.4	Métodos.....	13
3	PESQUISA BIBLIOGRAFICA	13
3.1	Museu Multitemático.....	13
3.2	Totem digital de interação	14
3.3	Software	15
3.3.1	IDE.....	16
3.3.2	Visual Studio.....	16
3.3.3	Trello.....	16
3.3.4	Figma.....	17
3.3.5	Github	17
3.3.6	Linguagem programação	17
3.3.7	.NET 7.0	18
3.3.8	Windows Forms.....	19
3.4	LGPD.....	19
3.4.1	Segurança da informação e dados dos clientes	20
3.4.2	Confidencialidade	20
3.4.3	Acesso mínimo necessário	20
3.4.4	Proibição de cópias não autorizadas	20
3.4.5	Transmissão segura de dados	20
3.4.6	Proteção física.....	21
3.4.7	Encerramento seguro.....	21
3.4.8	Conformidade com regulamentações	21
3.4.9	Aplicação LGPD no museu	21
4	DESENVOLVIMENTO	22
4.1	Diagramas	22
4.1.1	Diagrama de classe.....	22
4.1.2	Diagrama de caso de uso	24
4.1.3	Diagrama de sequência	25
4.2	Fluxogramas	26
4.3	Protótipo.....	29
4.4	Princípio central de qualidade do totem	29
4.4.1	Funcionalidades do ambiente do usuário	29
4.4.2	Usabilidade	29
4.4.3	Manutenibilidade	29
4.4.4	Portabilidade.....	30
4.5	Definição de requisitos funcionais	30

4.6	Definição de requisitos não funcionais.....	32
4.6.1	Desempenho	33
4.6.2	Confiabilidade.....	33
4.6.3	Segurança	34
4.6.4	Usabilidade	34
4.6.5	Manutenção.....	34
4.6.6	Eficiência	34
4.7	Definição de requisitos de negócio	34
4.7.1	Calculo do relatório final	36
4.8	Planejamento de Qualidade de Software (QS): Período e Versão	37
4.9	Interface do usuário.....	38
4.10	Metodologia.....	41
5	IMPLANTAÇÃO DO SISTEMA E TOTEM FÍSICO	43
5.1.1	Requisito mínimo indicado para utilização em totem de atendimento	43
5.1.2	Localização ideal para instalação dos totens	43
CONCLUSÃO.....		44
REFERÊNCIAS.....		45
APÊNDICE A – CÓDIGO FONTE C#		47
APÊNDICE B – MANUAL DO USUÁRIO		96
APÊNDICE C – CRONOGRAMA DE GRANTT		112

1 INTRODUÇÃO

Atualmente, o desenvolvimento de software está cada vez mais complexo, pois estão sendo utilizados paradigmas de programação diferentes. Um exemplo marcante desse avanço é a (P.O.O), que revoluciona a maneira como os programas são concebidos e implementados.

Além de se usar ferramentas e linguagens de programação que consiste para um desenvolvimento ainda mais efetivo, o projeto do totem no modelo ágil (*Scrum*) oferece estrutura e direções às equipes. Projetadas para promover a colaboração, transparência e adaptação contínua, essas práticas têm se tornado fundamentais para o sucesso de projetos em um ambiente ágil.

Nesse cenário, softwares estão presentes em praticamente todos os aspectos da vida cotidiana, oferece interatividade e facilita processos como avaliações e pesquisas de satisfação. Este projeto exemplifica como as tecnologias podem contribuir para a análise de qualidade e decisões sobre alterações de suas estruturas, produtos ou serviços oferecidos, no caso do museu que apresenta obras, assim possibilita o cliente avaliar e interagir com dispositivos de totens com perguntas pré-configuradas de forma estratégica.

O projeto apresenta desenvolvimento de um totem para o museu com o tema 'A primeira viagem do homem à lua', permitindo que os visitantes deem sua nota avaliativa fornecendo feedbacks. Nesse processo, um relatório de satisfação assertivo é gerado apontando onde pode-se melhorar as exposições, proporcionando assim uma experiência completa aos visitantes.

Além disso, será apresentadas telas do protótipo do software, demonstrando sua interface e layout. Essas telas são utilizadas como base para visualização e compreensão do funcionamento do software.

Ao longo deste trabalho, é abordado também aspectos como o planejamento da Qualidade de Software (QS) com base na ISO 9126, com definição de período e versão para garantir o controle e a melhoria contínua do produto até sua descontinuação.

2 OBJETIVOS

Desenvolver e apresentar um totem para um museu, com a finalidade de auxiliar seus visitantes e melhorar o funcionamento do local, de acordo com as opiniões e os níveis de satisfação deles no questionário.

Durante o processo, são utilizados cálculos para a criação de relatórios sobre o desenvolvimento da obra e sobre o tema, referente à “Primeira viagem do homem à lua”. Aplicando conhecimentos acadêmicos e respeitando as normas da LGPD para garantir segurança aos dados dos visitantes.

O projeto inclui a criação de um software que contenha aplicabilidade e utilidade para interatividade com os visitantes e melhorias no museu, a partir dos questionários de satisfação.

2.1 Objetivos Gerais

Tendo por base os conteúdos das disciplinas do atual semestre, o grupo do PIM deverá apresentar um projeto de um totem que auxilia os visitantes de um museu.

2.2 Objetivos específicos

- Desenvolver a capacidade de identificar as necessidades do usuário e traduzi-las com soluções;
- Argumentar e discutir as tecnologias utilizadas nos projetos de sistemas computacionais;
- Fomentar o hábito de trabalho em equipe e execução de projetos que envolva múltiplas disciplinas.

2.3 Recursos

Para o desenvolvimento do projeto, foram utilizados os seguintes recursos computacionais:

- Trello – Cronograma;
- Figma – Prototipação;

- Visual Studio – IDE;
- Discord e Teams – Comunicação da equipe;
- Pacote Microsoft Office – Edição e formatação do trabalho;
- Computador pessoal – Acesso aos softwares;
- Windows Forms – Informações;
- Lucidchart – Diagrama de Sequência.

2.4 Métodos

Incluiu pesquisas em fontes diversas, como sites dentro dos princípios das normas internacional e literatura acadêmica, bem como materiais e diálogos com professores especializados nas áreas correlatas e discussões com profissionais da tecnologia da informação. A junção desses métodos e recursos foi de extrema relevância na concepção e conclusão bem-sucedida.

3 PESQUISA BIBLIOGRAFICA

Para aprofundar o entendimento sobre o museu e sua estrutura foi fundamental explorar componentes que contribuem para uma visão mais abrangente.

3.1 Museu Multitemático

Segundo Bullos (2018) compreendendo a natureza dos museus multitemáticos, pode-se observar que eles se destacam pela apresentação de uma variedade de tópicos e disciplinas em suas exposições e coleções. Em contraste com os museus especializados, que se concentram em um único assunto, os museus multitemáticos oferecem uma ampla gama de temas, incluindo história, ciência, arte e cultura. Eles frequentemente alternam suas exposições, o que atrai um público igualmente diversificado. Um exemplo de museu multitemático é o "Museu Multitemático de Barra Mansa".

Esses museus desempenham um papel significativo em termos de educação, preservação do patrimônio e na oferta de experiências variadas para o público conforme a (Figura 1) ilustrativa a seguir com diversidade de temas.

Figura 1 Ilustração planta baixa museu multitemático



Fonte: Próprio autor

3.2 Totem digital de interação

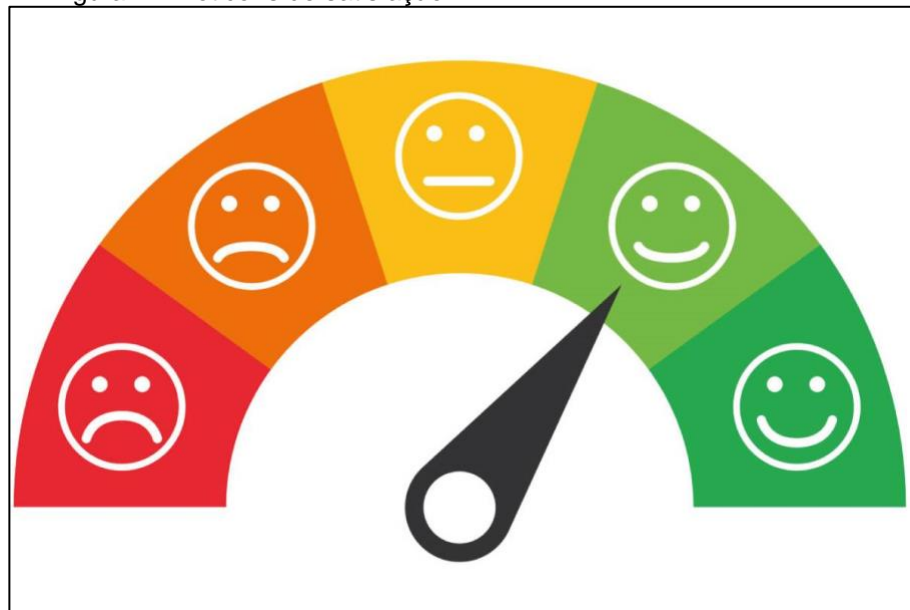
Com frequência, os gestores enfrentam a necessidade de compreender a percepção dos clientes em relação aos produtos ou serviços oferecidos pois surgem dúvidas sobre a satisfação, sugestões de melhoria, ou até mesmo reclamações. Em muitos casos, é desafiador interagir com os clientes de forma individual e eficiente. É nesse contexto que os totens digitais de interação com o cliente desempenham um papel fundamental, permitindo a coleta de feedback de forma prática e interativa. Essa tecnologia possibilita a obtenção de relatórios detalhados, contribuindo para a tomada de decisões embasadas e o aprimoramento contínuo dos produtos e serviços oferecidos pela empresa.

Segundo Saluceste (2019), Totens interativos ou quiosques são equipamentos com grandes monitores touch screen. Integrados a um software construído de forma intuitiva, com funções específicas para o toque, os equipamentos podem trazer grande atratividade e inovação ao serem utilizados, criando uma fantástica experiência para o usuário.

Combinar pesquisa de satisfação de qualidade com um totem interativo oferece uma maneira conveniente, eficiente e econômica de coletar feedback dos usuários e

melhorar continuamente a experiência, utilizando os emoticons conforme a (Figura 2) para demonstrar os sentimentos em uma tela interativa digital.

Figura 2 Emoticons de satisfação



SIPCES, (2021).

3.3 Software

O software consiste em um conjunto de instruções destinadas a orientar o computador na execução de tarefas específicas, abrangendo programas, dados e comandos essenciais para o seu funcionamento.

Software é uma sequência de instruções escritas para serem interpretadas por um computador para executar tarefas específicas. Também pode ser definido como os programas, dados e instruções que comandam o funcionamento de um computador (COELHO, 2016?).

Segundo Coelho (2016), software é um conjunto de instruções escritas onde um computador, smartphone, tablet ou outro dispositivo eletrônico segue para executar tarefas específicas. Este conjunto, que abrange programas, dados e comandos, atua como a "mente" do funcionamento da máquina, permitindo que ela realize as ações solicitadas pelo usuário. Enquanto o hardware é a parte física do dispositivo, o software representa a parte lógica e imaterial, fornecendo orientações vitais para o hardware operar de maneira eficiente.

3.3.1 IDE

Popularmente conhecido como IDE, o Ambiente Integrado de Desenvolvimento é uma ferramenta essencial na programação. Em essência, é um software que reúne várias funcionalidades necessárias para criar um aplicativo em uma única interface gráfica.

O ambiente de desenvolvimento integrado (IDE do inglês Integrated Development Environment), é uma ferramenta de desenvolvimento para editar o código, acessar um terminal, executar um script, debugar e compilar usando um único ambiente. Um IDE é um programa repleto de funcionalidades que podem ser usadas por muitos aspectos no desenvolvimento de software, que inclui ferramentas de preenchimento de código, plugins, e muitos outros recursos para facilitar o processo de desenvolvimento de software (LIRA, 2022).

3.3.2 Visual Studio

É uma ferramenta de desenvolvimento que realiza todos os processos e etapas em um único ambiente integrado. Oferece recursos essenciais para o desenvolvimento de software, como criação de código, edição, depuração e compilação. Com isso, o Visual Studio permite ainda mais eficiência nos procedimentos e fases do desenvolvimento de software.

De acordo com Anand et al. (2023), Visual Studio é uma ferramenta de desenvolvedor poderosa que permite realizar todo o ciclo de desenvolvimento em um só lugar. É um IDE (Ambiente de Desenvolvimento Integrado) abrangente que você pode usar para escrever, editar, depurar e criar código, além de implantar seu aplicativo. Além da edição e da depuração de código, o Visual Studio inclui compiladores, ferramentas de preenchimento de código, controle do código-fonte, extensões e muito mais recursos para aprimorar cada estágio do processo de desenvolvimento de software.

3.3.3 Trello

Usado para gerenciamento de projetos baseada na web que utiliza quadros (boards), listas e cartões para ajudar a organizar tarefas e colaborar em equipe. Cada quadro representa um projeto, as listas representam etapas ou categorias dentro do projeto, e os cartões representam tarefas específicas. É amplamente usado por equipes para planejar, rastrear e gerenciar trabalho, proporcionando uma visão clara e visual do progresso e do status das tarefas.

O Trello é uma ferramenta que oferece um plano gratuito para organizar desde nossas tarefas pessoais até demandas coletivas de uma equipe. Ele possibilita que criemos quadros com listas e adicionemos nelas cartões – ou cards, como costumamos dizer no dia a dia – com itens e tarefas, que são úteis para organizar times e atividades específicas para um determinado conjunto de pessoas (RIBEIRO, 2022).

3.3.4 Figma

Plataforma de criação de protótipos e design de interfaces, com recursos gratuitos. Utilizada para desenvolvimentos de telas, gráficos e gestão de projeto. Com isso o Figma é uma das soluções essenciais para *Squads* ou time que trabalham na metodologia ágeis.

O Figma é uma plataforma colaborativa para construção de design de interfaces e protótipos, pertencente a empresa Figma, Inc., lançada em 2016 por Dylan Field e Evan Wallace. O objetivo era o de criar uma ferramenta gratuita que trouxesse colaboração entre pessoas e times, permitindo criar um produto para as mais diversas plataformas, mantendo a acessibilidade do sistema (VILLAIN; SILVEIRA, 2023)

3.3.5 Github

Permite gerenciar projetos de códigos na nuvem, aplicando o sistema de controle de versões conhecido como Git, que possibilita aos desenvolvedores fazerem mudanças nos projetos compartilhados.

Conforme Fernanda e Louzada (2023) dizem que: "GitHub é uma plataforma para gerenciar seu código e criar um ambiente de colaboração entre devs, utilizando o Git como sistema de controle".

3.3.6 Linguagem programação

Em vez de os programadores se comunicarem com os computadores de maneira complexa, a linguagem de programação atua como uma ponte que simplifica essa comunicação. Elas funcionam como um tradutor entre a forma de pensar humana e a capacidade das máquinas de executar tarefas. Isso permite que os dispositivos ajam de acordo com as instruções definidas de maneira mais acessível e compreensível.

A expressão se refere a um conjunto de regras e instruções que um programador constrói para gerar programas e softwares que serão processados por um computador, dispositivo móvel ou outro equipamento (MENDES, 2022).

Conforme Mendes (2022), linguagem de programação é um conjunto de regras e instruções que os programadores usam para criar programas e *softwares*. Essas linguagens formam o código-fonte, que guia o comportamento de um computador ou dispositivo. Isso é preferido em relação ao código de máquina, pois torna a programação mais produtiva e compreensível, permitindo que os dispositivos ajam de forma mais independente e universal. As linguagens de programação consistem em símbolos, palavras-chave e regras que são classificadas em diferentes níveis, tornando-as uma parte essencial da tecnologia.

3.3.6.1 Linguagem C#

Conhecida por facilitar a produtividade dos desenvolvedores ao criar código eficiente e de alto desempenho. Conforme Wagner (2024) se trata da linguagem mais popular dentro do ambiente .NET, o que significa que é amplamente suportada e utilizada para uma variedade de tarefas e projetos. A linguagem C# é baseada na programação orientada a objetos, mas também inclui características de outros estilos de programação. Isso a torna flexível e capaz de lidar com diferentes tipos de problemas de programação. Muitas das ferramentas e bibliotecas do .NET são escritas em C#, e as melhorias na linguagem frequentemente trazem benefícios para todos os desenvolvedores que trabalham com o sistema. Isso faz do C# uma escolha poderosa e eficiente para desenvolvedores que trabalham nesse ecossistema.

3.3.7 .NET 7.0

De acordo com Warren (2024), o .NET é uma tecnologia que permite aos códigos desenvolvidos no Microsoft Visual Studio sejam compatíveis com quase todos os sistemas operacionais. Dessa forma, o software não fica restrito a uma única plataforma, podendo ser utilizado conforme as necessidades do cliente.

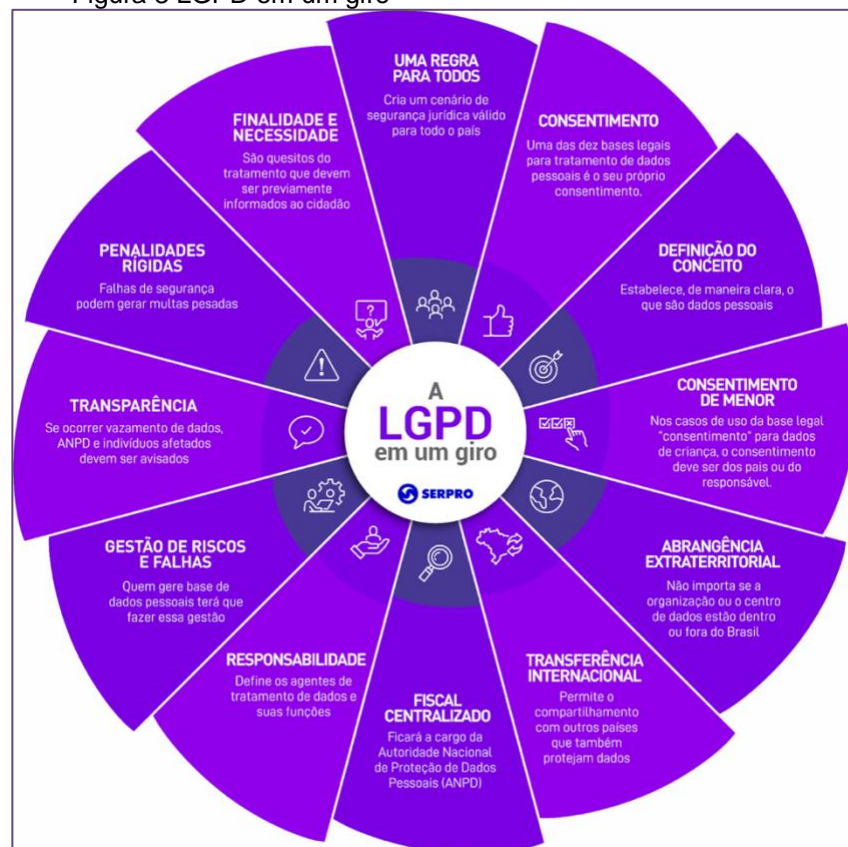
3.3.8 Windows Forms

Pela praticidade de ter uma tecnologia de criação de telas dentro da IDE utilizada facilita no desenvolvimento e na manutenção, além de ter tudo que uma tecnologia desse tipo deveria ter, conforme DOLLARD (2024).

3.4 LGPD

A Lei Geral de Proteção de Dados Pessoais (LGPD), Lei nº 13.709/2018, foi estabelecida com o propósito de preservar os direitos essenciais à liberdade, à privacidade e à construção livre da identidade de cada pessoa. As leis da LGPD apresentadas de forma simples na (Figura 3).

Figura 3 LGPD em um giro



Fonte: SERPRO.GOV, (2020)

3.4.1 Segurança da informação e dados dos clientes

A segurança dos dados dos clientes é uma responsabilidade essencial de todos os funcionários, incluindo os profissionais terceirizados. Além das medidas de segurança previamente mencionadas, é necessário seguir as diretrizes adicionais para proteção dos dados dos clientes da empresa.

3.4.2 Confidencialidade

Os dados dos clientes devem ser tratados como informações estritamente confidenciais. É proibido compartilhar informações dos clientes com pessoas não autorizadas, interna ou externamente, a menos que seja necessário para a execução das tarefas atribuídas.

3.4.3 Acesso mínimo necessário

Acesse apenas os dados dos clientes que sejam estritamente necessários para o desempenho das funções atribuídas. Evite visualizar ou acessar informações de clientes sem a devida autorização.

3.4.4 Proibição de cópias não autorizadas

Não faça cópias não autorizadas dos dados dos clientes. Isso inclui a cópia de informações confidenciais para dispositivos de armazenamento externos, como pendrives ou dispositivos pessoais.

3.4.5 Transmissão segura de dados

Sempre que necessário transmitir dados dos clientes, utilize métodos seguros, como criptografia ou sistemas de transferência de arquivos seguros (SFTP). Evite o uso de canais não seguros, como e-mails não criptografados, para compartilhar informações confidenciais.

3.4.6 Proteção física

Mantenha os dispositivos que contêm os dados dos clientes em locais seguros, protegidos contra roubo, acesso não autorizado ou danos físicos. Mantenha os dispositivos com acesso restrito, seja por meio de bloqueio físico ou por meio de senhas e autenticação adequadas.

3.4.7 Encerramento seguro

Ao concluir o trabalho ou encerrar o contrato de prestação de serviços, certifique-se de que todos os dados dos clientes sejam devidamente removidos ou excluídos dos dispositivos e sistemas utilizados. Siga as políticas e procedimentos estabelecidos para o descarte seguro de informações confidenciais.

3.4.8 Conformidade com regulamentações

Cumpra todas as regulamentações e leis aplicáveis relacionadas à privacidade e proteção de dados dos clientes. Isso inclui, por exemplo, a Lei Geral de Proteção de Dados (LGPD) no Brasil ou o Regulamento Geral de Proteção de Dados (GDPR) na União Europeia.

O não cumprimento dessas diretrizes de segurança de dados dos clientes pode resultar em ações disciplinares e possíveis consequências legais. É fundamental garantir a privacidade e a segurança dos dados dos clientes, protegendo sua confidencialidade e prevenindo o acesso não autorizado ou o uso indevido das informações.

3.4.9 Aplicação LGPD no museu

Em um totem de museu, a Lei Geral de Proteção de Dados (LGPD) é relevante principalmente no que diz respeito à coleta e ao tratamento de dados pessoais dos visitantes. Isso pode incluir informações como nome, idade, e outras informações pessoais que os visitantes possam fornecer ao interagir com o totem, como registros de entrada ou feedback sobre exposições. Para cumprir a LGPD, o museu deve garantir que os dados sejam coletados e processados de forma transparente, com

consentimento dos visitantes, e protegidos contra acesso não autorizado ou uso indevido.

4 DESENVOLVIMENTO

A elaboração do projeto se baseia nas informações fornecidas por outros processos, juntamente com a estratégia de planejamento, para criar um documento claro e unificado que orienta a execução e o acompanhamento. Esse processo é geralmente realizado múltiplas vezes.

4.1 Diagramas

Para descrever os requisitos funcionais de um sistema, representar a estrutura estática detalhando as classes, seus atributos, métodos, relacionamentos entre elas e a modelagem da interação entre diferentes objetos, são necessários os diagramas de classes, de casos de uso e de sequência.

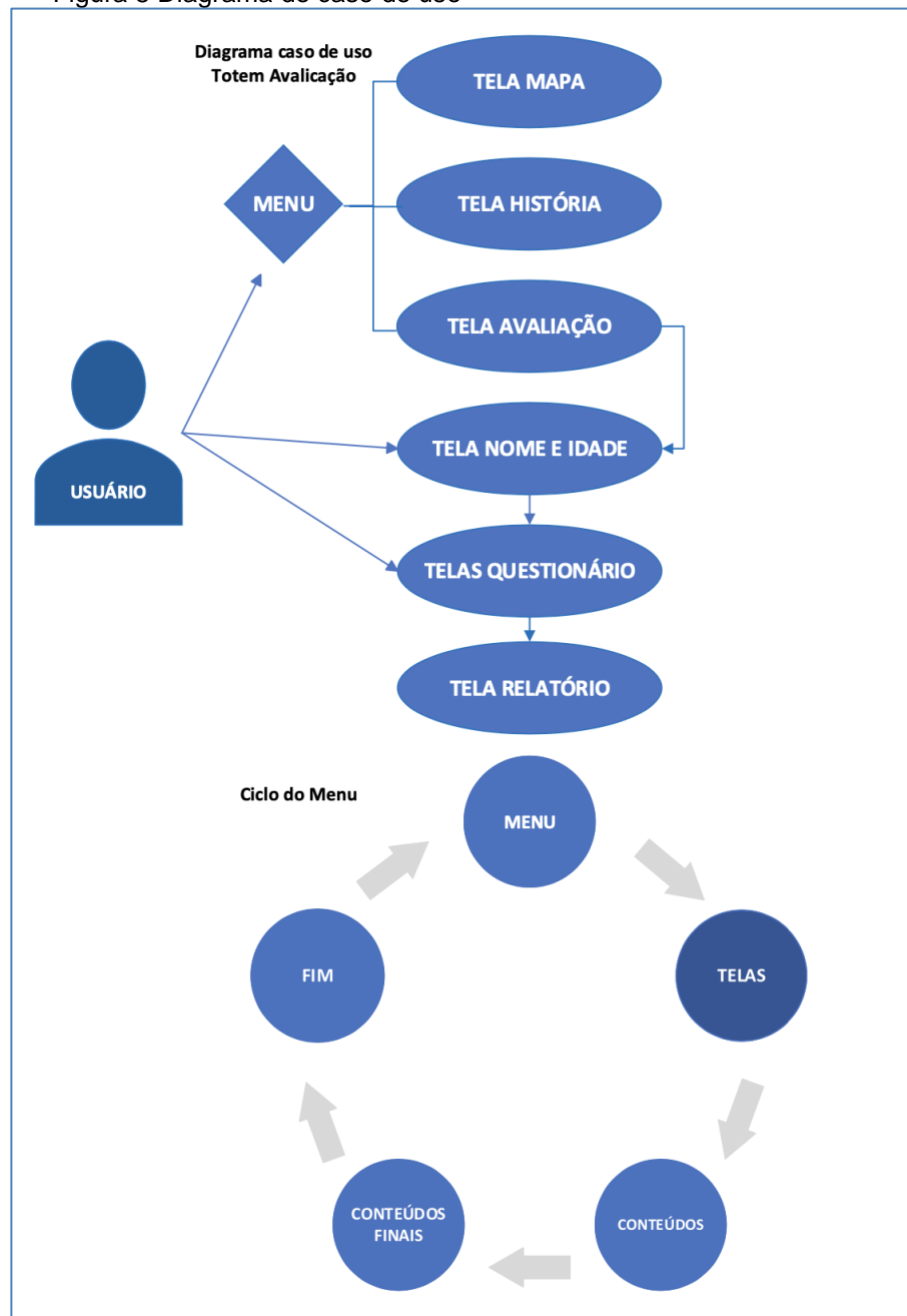
4.1.1 Diagrama de classe

O diagrama de classe tem como objetivo principal representar a estrutura estática do sistema, mostrando as classes, seus atributos, métodos e os relacionamentos entre elas. Ao entender esse diagrama possibilitou definir a arquitetura do sistema conforme a ilustração.

4.1.2 Diagrama de caso de uso

Por sua representação visual das interações entre o usuário e um sistema de software, o diagrama de caso de uso mostra seus relacionamentos úteis. Sendo fundamental para entendimento da equipe compreender a comunicação com clareza para todos da equipe de desenvolvimento.

Figura 5 Diagrama de caso de uso

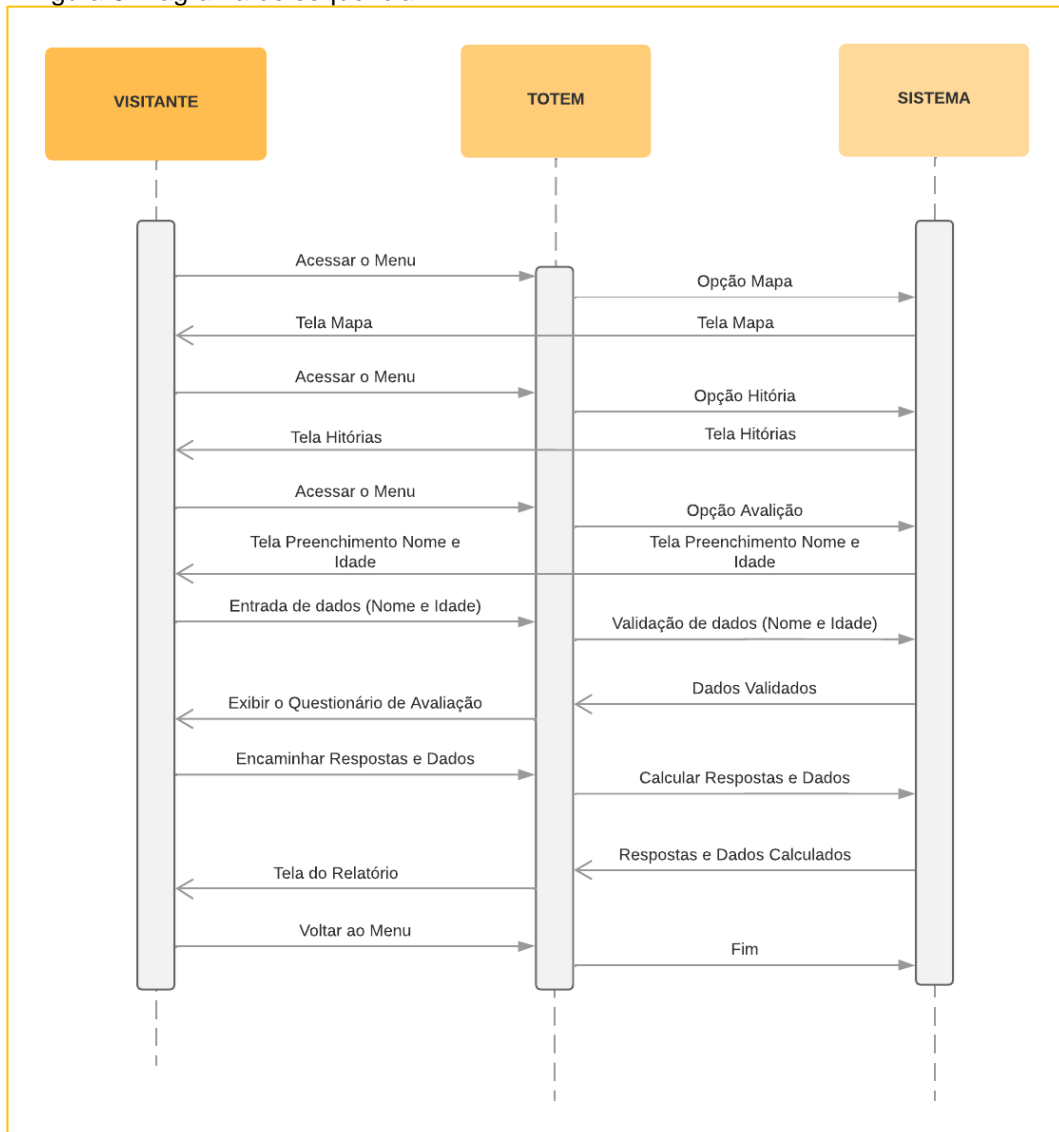


Fonte: Próprio autor

4.1.3 Diagrama de sequência

O objetivo deste diagrama é determinar a comunicação dos processos dentro do fluxo do sistema. Os atores e atividades estão ligados ao caso de uso mencionado na (Figura 6). O diagrama possui dois eixos, o vertical indica a sequência das mensagens e o tempo de vida, enquanto o eixo horizontal refere-se à linha de vida.

Figura 6 Diagrama de sequência



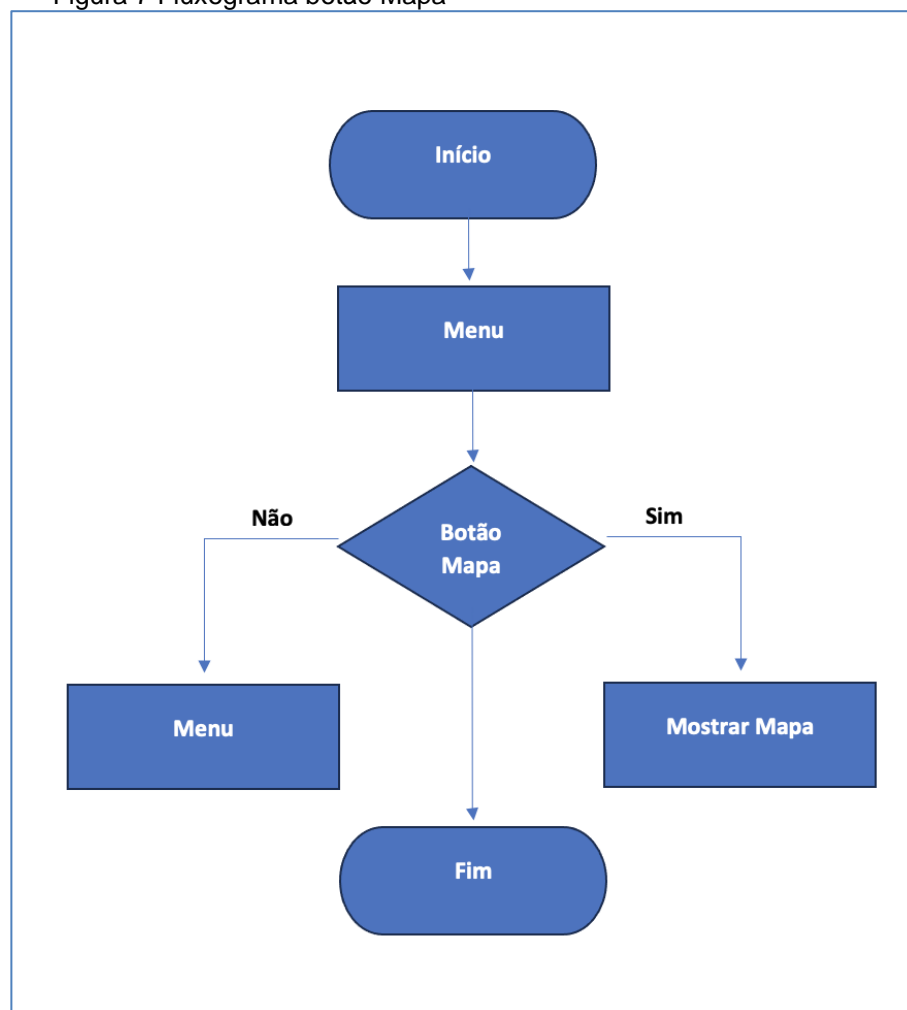
Fonte: Próprio autor

4.2 Fluxogramas

Para detalhar o processo do software, facilitar a compreensão da estrutura do funcionamento do sistema e melhorar a comunicação entre os membros da equipe de desenvolvimento e outras partes interessadas, estes fluxogramas visualizam cada etapa, desde a ação inicial do usuário, permitindo uma compreensão clara das interações e do fluxo de dados no sistema.

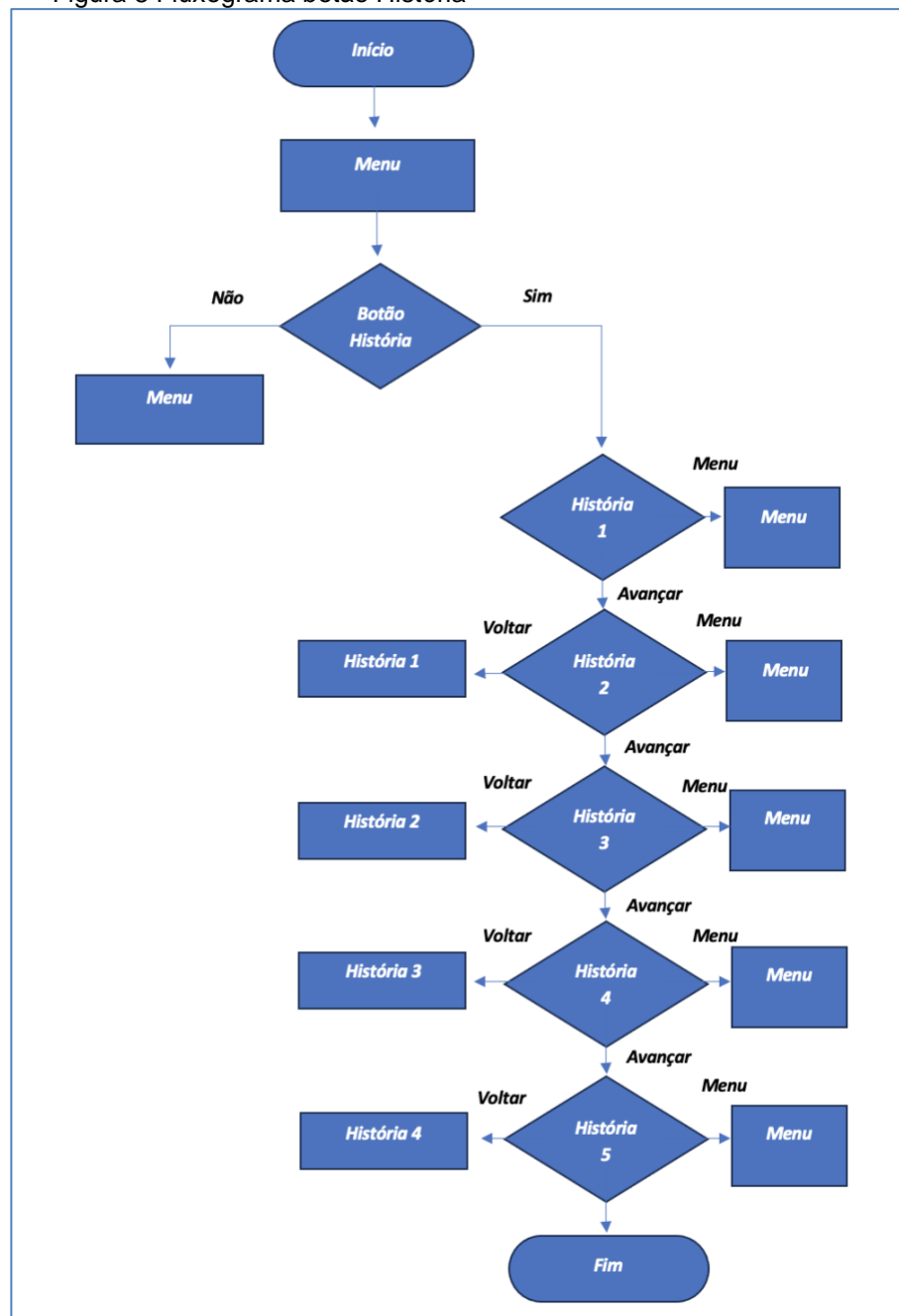
Na sequencia abaixo apresentam os fluxogramas do software com opções do menu inicial contendo os três botões (mapa, história e avaliação).

Figura 7 Fluxograma botão Mapa



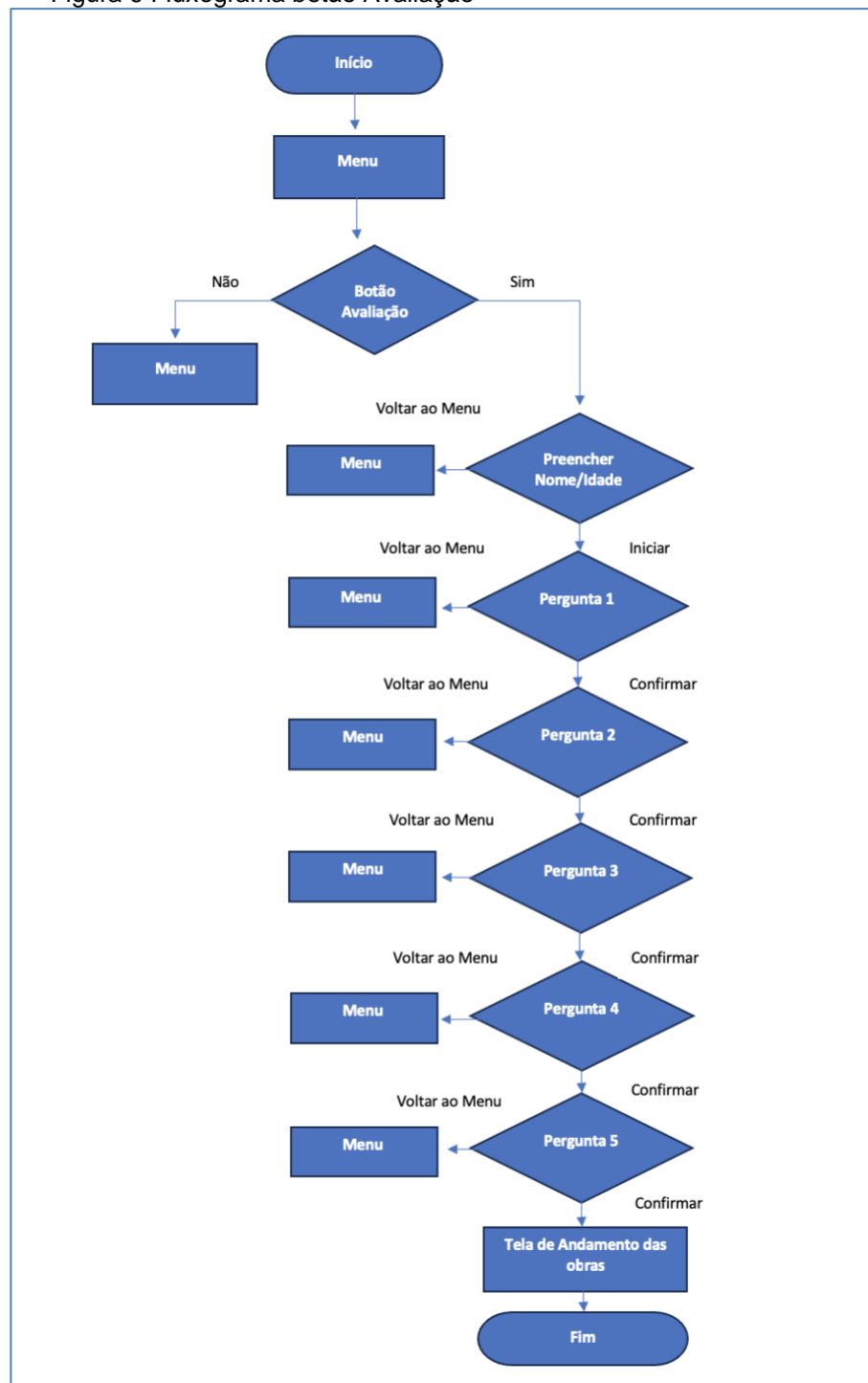
Fonte: Próprio autor

Figura 8 Fluxograma botão História



Fonte: Próprio autor

Figura 9 Fluxograma botão Avaliação



Fonte: Próprio autor

4.3 Protótipo

Apresentar um protótipo antes do desenvolvimento do código é essencial na metodologia ágil escolhida, pois permite a validação rápida dos requisitos e a obtenção de feedback imediato, facilitando ajustes precoces. Isso economiza tempo e recursos, reduzindo retrabalho e promovendo um alinhamento claro das expectativas entre a equipe. Além disso, melhora a usabilidade através de testes iterativos, facilita o planejamento das iterações, garantindo que as necessidades sejam continuamente atendidas.

4.4 Princípio central de qualidade do totem

O totem foi desenvolvido para garantir uma ótima experiência e eficiência para os usuários finais, com operações intuitivas e suporte técnico contínuo.

4.4.1 Funcionalidades do ambiente do usuário

O totem será disponibilizado para o cliente de forma física em uma tela touchscreen, contendo o menu inicial com três botões centralizados, um teclado virtual para preenchimento de dados, e por fim, cinco alternativas de avaliação em um quiz interativo.

4.4.2 Usabilidade

O software é atrativo para que o usuário final venha a responder ao questionário de sua experiência no local, além de ter uma interação fácil com ele. Com um layout simplificado com fontes de melhor posicionamento, tamanho e tipo, alternativas que são padrões utilizadas por várias empresas, apresentação de forma básica e simplificada, utiliza opções como “muito ruim, ruim, regular, bom e muito bom”.

4.4.3 Manutenibilidade

Todos os temas e perguntas presentes no software serão sempre atualizados conforme a necessidade do cliente, permitindo que ele avalie ou apresente o que for

necessário para receber um relatório e tomar decisões mais coerentes na melhoria do seu negócio.

Destaca a importância da manutenibilidade no desenvolvimento de software, enfatiza a necessidade de sistemas facilmente modificáveis para garantir sua eficácia ao longo do tempo, por conseguir atender as necessidades dos clientes.

4.4.4 Portabilidade

Como o software é criado em .NET 7.0 ele é compatível com a maioria dos sistemas operacionais hoje no mercado e como seu código fonte é bem otimizado ele utiliza pouco do processamento do computador.

4.5 Definição de requisitos funcionais

Conforme o que foi apresentado no escopo, discutido pela equipe do projeto e análise cuidadosa dos requisitos e das necessidades identificadas, foi declarado como requisitos funcionais:

- Tela sensível ao toque (touchscreen);
- Questionário a ser elaborado pelo grupo do PIM com 5 questões;
- Teclado virtual (desenvolvido pela equipe);
- O software será capaz de gerar um relatório por meio de cálculos com os dados coletados.

Quadro 1 RF Tela Touch

Identificador	RF001
Nome:	Tela sensível ao toque (touchscreen)
Módulo:	Tela apresentada para os clientes
Data da criação:	27/02/2024
Data da última modificação:	28/04/2024
Versão:	N/C
Descrição	O meio com que o visitante vai interagir com o totem é por meio de uma tela sensível ao toque

Fonte: Próprio autor

Quadro 2 RF Questionário

Identificador	RF002
Nome:	Questionário a ser elaborado pelo grupo do PIM com 5 questões
Módulo:	Tela apresentada para os clientes
Data da criação:	27/02/2024
Data da última modificação:	06/04/2024
Versão:	N/C
Descrição	As perguntas serão para avaliar a experiência do visitante no museu e com a exposição

Fonte: Próprio autor

Quadro 3 RF Teclado virtual

Identificador	RF003
Nome:	Teclado virtual (desenvolvido pela equipe)
Módulo:	Tela apresentada para os clientes
Data da criação:	27/02/2024
Data da última modificação:	11/05/2024
Versão:	N/C
Descrição	O software não poderá usar o teclado fornecido pelo sistema operacional da máquina, ele deverá ser desenvolvido pelos criadores do software.

Fonte: Próprio autor

Quadro 4 RF Relatório

Identificador:	RF004
Nome:	O software será capaz de gerar um relatório por meio de cálculos com os dados coletados
Módulo:	Tela apresentada para os clientes
Data da criação:	27/02/2024
Data da última modificação:	11/05/2024
Versão:	N/C
Descrição	Apresenta: Quantidade de perguntas respondidas; Total de usuários; Média Literária; Faixa etária (-18, +18 e +60); Média da avaliação.

Fonte: Próprio autor

4.6 Definição de requisitos não funcionais

Para garantir que o sistema atenda a critérios importantes relacionados à qualidade, desempenho e usabilidade, foram utilizados os softwares seguros, eficiente e confiável em seu funcionamento.

Foi feita a utilização da linguagem C# pela sua interatividade com o .NET e com o Microsoft Visual Studio, além de ser possível trabalhar com ela utilizando o paradigma de Programação Orientada a Objeto (POO). O Windows Forms foi escolhido como ferramenta de requisito não funcional devido à sua facilidade na criação de interfaces gráficas ricas e sua integração com o Visual Studio.

Quadro 5 RNF Linguagem C#

Identificador:	RNF001
Nome:	Linguagem C#
Módulo:	Software
Data da criação:	27/02/2024
Data da última modificação:	27/02/2024
Versão:	N/C
Descrição	Todo o código fonte escrito em linguagem c#.

Fonte: Próprio autor

Quadro 6 RNF Windows Forms

Identificador:	RNF002
Nome:	Windows Forms
Módulo:	Software
Data da criação:	27/02/2024
Data da última modificação:	27/02/2024
Versão:	N/C
Descrição	A parte de interface que aparecera na tela para o visitante será que ser desenvolvido usando a funcionalidade do Windows Forms.

Fonte: Próprio autor

Quadro 7 RNF Linguagem C#

Identificador:	RNF003
Nome:	.NET 7
Módulo:	Software
Data da criação:	27/02/2024
Data da última modificação:	27/02/2024
Versão:	N/V
Descrição	Por questão de compatibilidade foi utilizado a versão .NET 7 por ser compatível com a maioria dos Sistemas Operacionais.

Fonte: Próprio autor

4.6.1 Desempenho

O totem desenvolvido é capaz de lidar com vários usuários simultaneamente, garantindo tempos de resposta rápidos para as avaliações.

4.6.2 Confiabilidade

É um totem estável e confiável, minimizando o tempo de inatividade e garantindo que as avaliações dos visitantes sejam registradas com precisão.

4.6.3 Segurança

Os dados dos usuários/visitantes são protegidos e as informações coletadas durante as avaliações são mantidas confidenciais e seguras.

4.6.4 Usabilidade

A interface do totem é intuitiva e fácil de usar para os usuários/visitantes, com instruções claras e uma experiência agradável para o usuário final.

4.6.5 Manutenção

É importante destacar que o totem é fácil de manter e atualizar, com ferramentas de administração que são simples no gerenciamento do sistema e garantindo sua integridade ao longo do tempo.

4.6.6 Eficiência

O totem utiliza recursos de hardware e software de forma eficiente, minimizando o consumo de energia e maximizando o desempenho durante as avaliações.

4.7 Definição de requisitos de negócio

O projeto do totem deve oferecer informações concisas e relevantes sobre as exposições e obras referente ao tema da Primeira Viagem do Homem à Lua. Isso inclui a capacidade de navegação intuitiva para os visitantes avaliarem as exposições e o atendimento no museu. Além disso, é fundamental que o totem seja visualmente atraente e de fácil utilização para pessoas de todas as idades e condições.

Para contribuindo para o sucesso e a sustentabilidade da organização através da melhoria contínua e do foco no cliente dentro dos princípios das ISO de qualidade abaixo:

- ISO 9001 (Sistema de gestão da qualidade);
- ISO 25000 (Requisitos e avaliação de qualidade de sistemas e software);

- ISO 27001 (Sistema de gestão de segurança da informação);
- ISO 12207 (Processos de ciclo de vida do software);
- ISO 9126 (Qualidade de produto de software);
- ISO 14598 (Documentação de módulos de avaliação);
- ISO 20252 (Requisitos específicos e técnicos).

No caso de tratamento de pesquisas qualitativas ou quantitativas mesmo que os dados sejam colhidos por totem de atendimento leva em consideração os dados mesmo que não pessoais atendendo as ISO 20252.

Para garantir a conformidade com a LGPD, os questionários são estruturados para não coletar informações sensíveis dos visitantes. O texto é apresentado aos usuários na tela inicial com a seguinte descrição: “Nós respeitamos sua privacidade e seguimos as normas da LGPD, mesmo tratando apenas de dados não pessoais. Sua proteção é nossa prioridade”.

Além disso, são programados seguindo os princípios de Projetos de Interface com o usuário para uma experiência intuitiva.

Quadro 8 RN Utilização de ISO

Identificador	RN001
Nome:	Utilização de ISO
Módulo:	Normas
Data da criação:	20/04/2024
Data da última modificação:	19/05/2024
Versão:	0.2
Descrição	<p>Seguir o padrão que se pede nas normas ISO</p> <p>(9001) (14598)</p> <p>(25000) (20252)</p> <p>(27001)</p> <p>(12207)</p> <p>(9126)</p>

Fonte: Próprio autor

Quadro 9 RN LGPD

Identificador	RN002
Nome:	LGPD
Módulo:	Normas
Data da criação:	27/02/2024
Data da última modificação:	27/02/2024
Versão:	0.1
Descrição	O software não poderá coletar dados sensíveis dos visitantes.

Fonte: Próprio autor

Quadro 10 RN Home a lua

Identificador	RN003
Nome:	Homem a lua
Módulo:	Negocio
Data da criação:	27/02/2024
Data da última modificação:	27/02/2024
Versão:	0.1
Descrição	O tema da exposição onde o software será usado vai ser sobre o homem a lua, portanto toda sua temática deverá ter esse parâmetro.

Fonte: Próprio autor

Em resumo, os requisitos do negócio para um totem em um museu devem priorizar a acessibilidade, usabilidade e confiabilidade para oferecer uma experiência inesquecível aos visitantes.

4.7.1 Cálculo do relatório final

Para apresentar o relatório final das pesquisas de qualidade, faz-se uso do cálculo CES 1.0. Esse método é essencial porque mede a facilidade com que os

clientes interagem com o sistema, proporcionando uma métrica objetiva que reflete diretamente a experiência de uso e a satisfação.

A (Tabela 1) apresenta como é feito o cálculo do CES 1.0, utilizando a fórmula de média ponderada, que multiplica a quantidade de escolhas (respostas) de cada opção (1, 0, 3, 11, 5) pelo seu peso de 1 a 5, posteriormente somando o total e dividindo pelo total das respostas (20). Apresentando a seguinte forma $CES = (1 \times 1) + (2 \times 0) + (3 \times 3) + (4 \times 11) + (5 \times 5) \div 20 =$

$$1 + 0 + 9 + 44 + 25 \div 20 =$$

$$79 \div 20 =$$

Valor apresentado no relatório: 3,95 (BOM).

Tabela 1 Cálculo CES 1.0

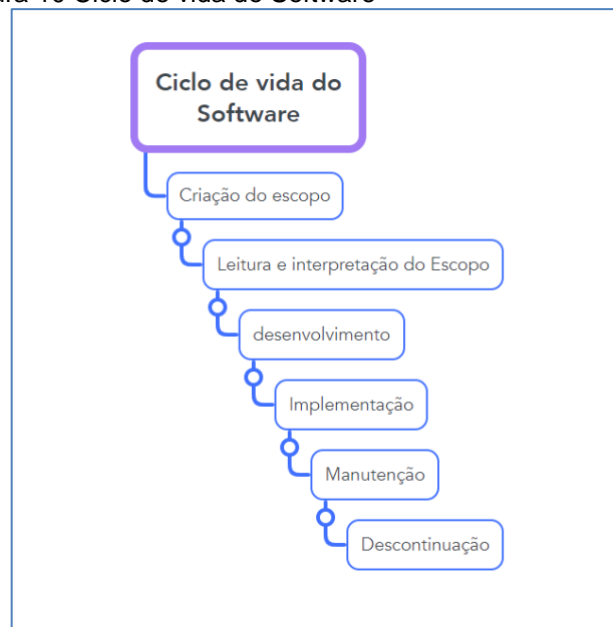
Cálculo do CES 1.0				
Pessoas		4	T.PxR/T.RESP	
PESO		PESOxRESP	Divisão média 3,95 Bom	
1	Muito ruim	1	se menor que 2 mostrar "Muito ruim"	
2	Ruim	0	se não, se menor que 3 mostrar "Ruim"	
3	Regular	3	se não, se menor que 4 mostrar "Regular"	
4	Bom	11	se não, se menor que 5 mostrar "Bom"	
5	Muito bom	5	se não mostra "Muito bom"	
Total respostas		20		
		79		

Fonte: Próprio autor

4.8 Planejamento de Qualidade de Software (QS): Período e Versão

Como o software do totem será usado indefinidamente, será garantido suporte contínuo à empresa que adquirirá o produto. Isso incluirá atualizações com alto esquema de segurança, correção de bugs e implementação de novos recursos, conforme necessário. Assim, também assistência máxima para os usuários finais. Integrando todos esses elementos no ciclo do software do totem, para poder garantir sua durabilidade e eficácia ao longo do tempo. Isso proporcionará à empresa compradora do totem uma experiência contínua e confiável, bem como para o usuário final. A seguir na (Figura 10) está um digrama geral do ciclo de vida do software.

Figura 10 Ciclo de vida do Software



Fonte: Próprio autor

4.9 Interface do usuário

A interface do usuário do Software do Totem é composta pelos seguintes elementos principais.

Tela principal: Onde os usuários podem acessar todas as funcionalidades do software, incluindo Mapa, Histórias e Avaliação.

Figura 11 Tela Principal



Fonte: Próprio autor

Tela Avaliação: Permite aos usuários avaliar as obras expostas no museu com 5 perguntas de avaliação

Figura 14 Tela Avaliação



Fonte: Próprio autor

Tela relatório: Permite aos usuários visualização de relatórios geral e cálculos de métricas.

Figura 15 Tela Relatório



Fonte: Próprio autor

4.10 Metodologia

A equipe optou pelo método ágil de desenvolvimento de software para promover uma abordagem colaborativa e flexível. Cada membro foi designado para trabalhar em partes específicas do projeto. Foram realizadas reuniões regulares para acompanhar o progresso, discutir possíveis melhorias e garantir que estivessem alinhadas com os requisitos do cliente.

Colaboração e comunicação constantes entre os membros da equipe e o cliente. Isso foi alcançado por meio de reuniões regulares para evolução do projeto, discussões abertas para testar, alterar ou remover funcionalidades e o envolvimento de todos os membros da equipe nas decisões e melhorias do processo de desenvolvimento.

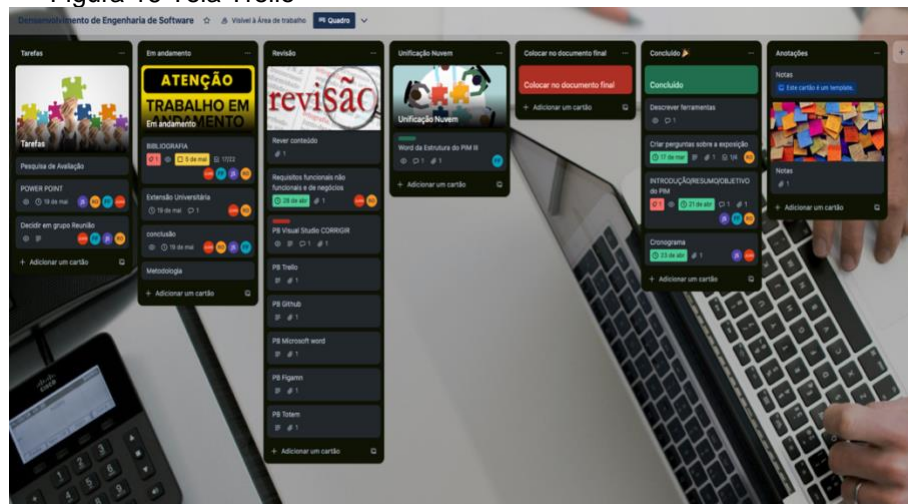
Entrega incremental e feedback rápido do software foram adotados, permitindo que o cliente tivesse acesso a partes do produto funcional mais cedo. Isso ajudou a validar de acordo com os requisitos e expectativas.

Adaptação às mudanças de requisitos do projeto foi realizada. Ao desenvolver partes do projeto em pequenos ciclos de trabalho (sprints), a equipe pôde ajustar e modificar conforme necessário, mantendo o produto alinhado com as necessidades do cliente.

A validação por protótipo foi apresentada para identificar se todas as funções estavam de acordo com a documentação e requisitos obrigatórios. Isso permitiu que a equipe e o cliente identificassem rapidamente qualquer discrepância ou necessidade de ajuste antes da implementação completa do software.

Para auxiliar nesse processo, a ferramenta Trello possuía campos de desenvolvimento divididos em partes e com distribuição responsabilidades, permitindo que os membros acompanhassem o andamento de cada etapa. Além disso, havia um campo exclusivo chamado “revisão” para validar se o trabalho realizado por outro membro estava correto antes de ser incorporado ao projeto.

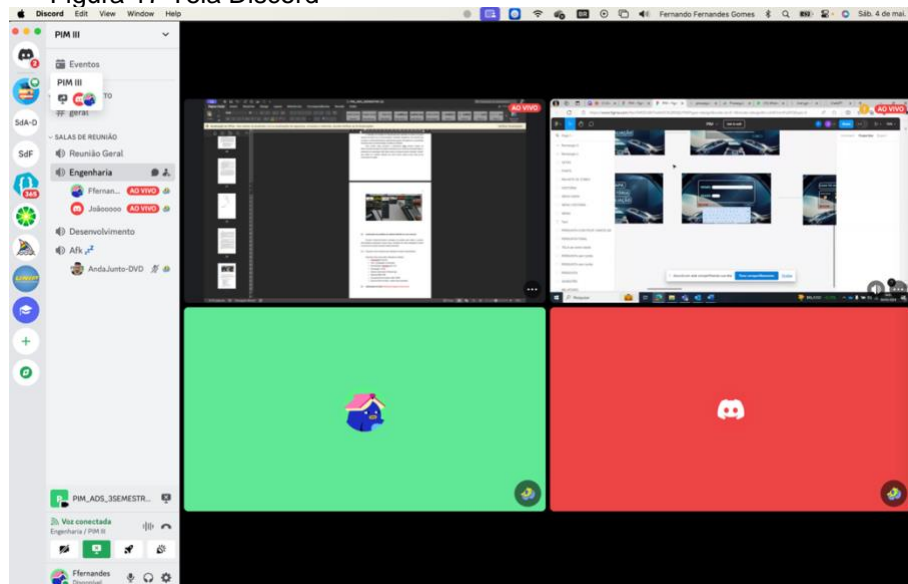
Figura 16 Tela Trello



Fonte: Próprio autor

O Discord foi importante para o trabalho porque facilitou a comunicação, organizou as atividades e permitiu a colaboração à distância de forma prática e integrada. Além disso, criou um ambiente descontraído para interação entre os membros da equipe possibilitando produtividade da equipe.

Figura 17 Tela Discord



Fonte: Próprio autor

5 IMPLANTAÇÃO DO SISTEMA E TOTEM FÍSICO

Alguns requisitos devem ser atendidos para o funcionamento adequado do sistema, como o local de instalação e o hardware do totem, para que haja o melhor aproveitamento possível.

5.1.1 Requisito mínimo indicado para utilização em totem de atendimento

Dispositivo físico para melhor utilização do software.

- Computador All In One
- Tela: 15 polegadas Touchscreen
- Marca/Modelo: Bematech SB-1015
- Processador: J4125
- Sistema Operacional: Windows IoT
- Memória RAM: 4GB
- Armazenamento de dados: SSD 120GB
- Estrutura física do totem: a definir pelo proprietário

5.1.2 Localização ideal para instalação dos totens

Recomenda-se instalar o totem no ambiente principal do museu, permitindo que os visitantes avaliem de forma imediata após experienciar as exposições, demonstrando compromisso com a melhoria contínua e fortalecendo a confiança.

CONCLUSÃO

O presente trabalho analisou os fundamentos do desenvolvimento de um software para a criação de um totem de avaliação, incluindo desde sua concepção até sua possível implementação em uma empresa.

Com o objetivo de proporcionar uma avaliação rápida e intuitiva para os visitantes, foi criado um software de satisfação, que conta com relatórios, baseado em todas as informações registradas pelos usuários.

Por meio da metodologia *Scrum* ágil adotada, foi possível definir os requisitos funcionais e não funcionais, além de permitir o gerenciamento de todas as atividades e o planejamento da Qualidade de Software para garantir a eficiência e a satisfação do cliente final.

A apresentação das telas do protótipo do software proporcionou uma visão prática de sua interface e layout, enquanto a discussão sobre sua implementação em uma empresa destacou a relevância e os benefícios concretos que essa solução pode proporcionar.

Com base nas matérias de Análise de Sistema Orientado a Objeto, Banco de Dados, Engenharia de software, Projeto de Interface com Usuário, Gestão estratégia de RH, Economia e Mercado e Programação Orientada a Objeto, foi possível desenvolver um software acessível, intuitivo e projetado para atender as expectativas do cliente.

Por fim, este trabalho contribuiu para o avanço do conhecimento na área de desenvolvimento de software, com atividades similares ao mercado de trabalho, e para a melhoria contínua dos serviços de atendimento ao público, promovendo uma experiência mais satisfatória, tanto para os clientes, quanto para as empresas.

REFERÊNCIAS

ANAND, Megha. et. al. **O que é o Visual Studio**. Microsoft, 2023. Disponível em: <https://learn.microsoft.com/pt-br/visualstudio/get-started/visual-studio-ide?view=vs-2022>. Acesso em: 27 abr. 2024.

BULLOS, Francis. **Museu multitemático**. Foco Regional, 2018. Disponível em: <https://www.focoregional.com.br/Noticia/museu-particular-mostra-seu-acervo-a-fundacao>. Acesso em: 27 abr. 2024.

DOLLARD, Kathleen. **Noções básicas de Aplicativo do Windows Forms (Visual Basic)**. Microsoft, 2024. Disponível em: <https://learn.microsoft.com/pt-br/dotnet/visual-basic/developing-apps/windows-forms/>. Acesso em: 27 abr. 2024.

FERNANDA, Camila; LOUZADA, Vinicius. **GitHub**. Alura 2023. Disponível em: <https://www.alura.com.br/artigos/o-que-e-git-github#o-que-e-github>. Acesso em: 27 abr. 2024.

LIRA, Guilherme, **Saiba tudo sobre o IDE - Integrated Development Environment**, Alura, 2022. Disponível em: <https://www.alura.com.br/artigos/o-que-e-uma-ide>. Acesso em: 27 abr. 2024

MENDES, Tatyane. **Linguagem de programação: Conheça os principais conceitos e ferramentas desse universo**. Na Pratica, 2022. Disponível em: <https://napratica.org.br/linguagem-de-programacao/>. Acesso em: 26 out. 2023.

PADILHA, Adriano. Et. al. **Software**. Enciclopédia Significados, 2016?. Atualizado em 2023. Disponível em: <https://www.significados.com.br/software/>. Acesso em: 27 abr. 2024.

RIBEIRO, Andre. **Trello. Conhecendo a ferramenta**. Alura 2022. Disponível em: <https://www.alura.com.br/artigos/trello>. Acesso em: 27 abr. 2024.

SALUCESTE, Gian Carlo. **Totens interativos: Uma solução digital para aprimorar**. LinkedIn, [S.l.], 20 fevereiro 2019. Disponível em: https://pt.linkedin.com/pulse/totens-interativos-uma-solu%C3%A7%C3%A3o-digital-para-aprimorar-gian-saluceste?trk=public_profile_article_view. Acesso em: 28 abr. 2024.

SERVIÇO FEDERAL DE PROCESSAMENTO DE DADOS. **Emoticons de satisfação**. SERPRO, 2020. Imagem digital. Disponível em <https://www.serpro.gov.br/lgpd/noticias/2020/lgpd-giro>. Acesso em: 13 abr. 2024

SINDICATO PATRONAL DE CONDOMÍNIOS. **A LGPD em um giro**. SIPCES, 2021. Imagem digital. Disponível em: <https://www.sipces.org.br/materias,20674,associados-respondem-pesquisa-satisfacao-sipces.html>. Acesso em: 13 abr. 2024.

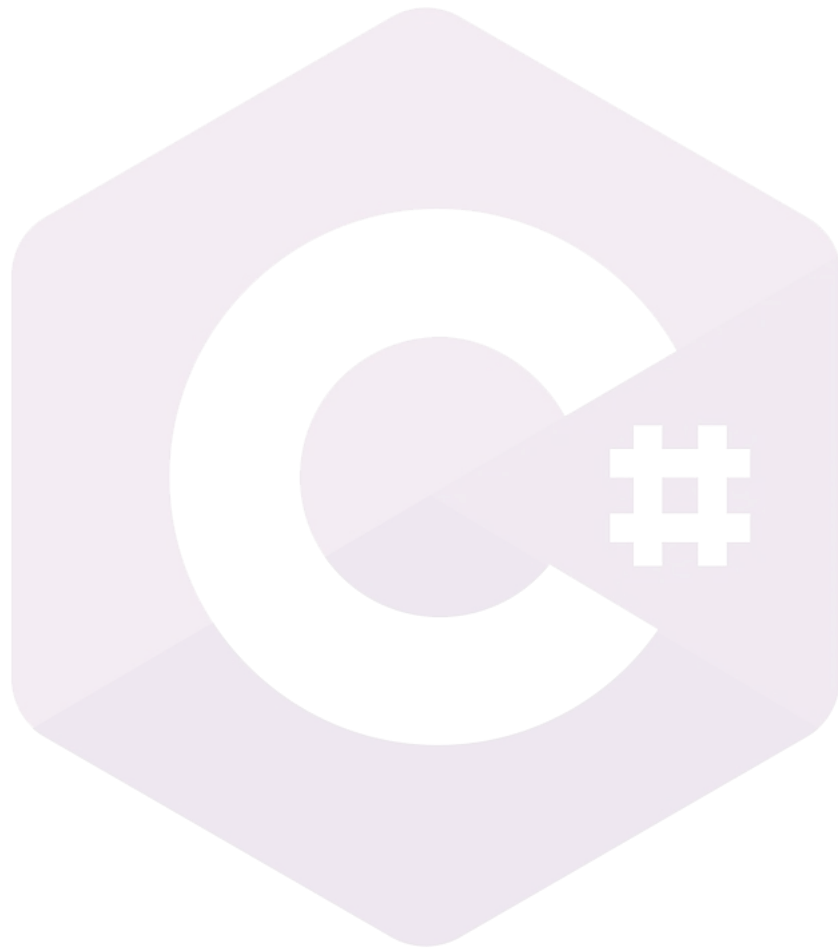
VILLAIN, Mateus; SILVEIRA, Maria Isabelle. **Figma — O que é?** Alura, 2023. Disponível em: <https://www.alura.com.br/artigos/figma>. Acesso em: 28 abr. 2024.

WAGNER, Bill. **Um tour pela linguagem C#**. Microsoft, 2024. Disponível em: <https://learn.microsoft.com/pt-pt/dotnet/csharp/tour-of-csharp/overview>. Acesso em: 20 mai. 2024.

WARREN, Genevieve. **Introdução ao .NET**. Microsoft, 2024. Disponível em: https://learn.microsoft.com/pt-br/dotnet/core/introduction?WT.mc_id=dotnet-35129. Acesso em: 27 abr. 2024.

APÊNDICE A – CÓDIGO FONTE C#

PROJETO DE UM TOTEM DE MUSEU MULTITEMÁTICO



CLASSE ABSPROPRIEDADES

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Security.Policy;
using System.Text;
using System.Threading.Tasks;

namespace SoftwarePIM.Back_end
{
    public abstract class Abspropriedades
    {
        public int alternativa;
        public int numero_pergunta;
        public int muito_ruim;
        public int ruim ;
        public int regular;
        public int bom;
        public int muito_bom;
        public string nome = "";
        public string idade = "";
        public string mensagem = "ok";

    }
}
```

CLASSE CONTROLE

```
using SoftwarePIM.Back_End;
using SoftwarePIM.Modelo;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace SoftwarePIM.Back_end

{
    internal class Controle : Abspropriedades
    {

        public Controle(int muito_ruim, int ruim,int regular,int bom,int muito_bom, int
numero_pergunta)

        {
            this.muito_ruim = muito_ruim;
            this.ruim = ruim;
            this.regular = regular;
            this.bom = bom;
            this.muito_bom = muito_bom;
            Executar_questionario();
        }

        public Controle(string nome, string idade)
        {
            this.nome = nome;
            this.idade = idade;
            Executar_validacao();
        }
    }
}
```

```
}
```

```
private void Executar_questionario(){
```

```
    Relatorio relatorio = new Relatorio(this.muito_ruim, this.ruim, this.regular,
this.bom, this.muito_bom);
```

```
}
```

```
private void Executar_validacao()
```

```
{
```

```
    Validacao validacao = new Validacao(nome,idade);
```

```
    this.mensagem = validacao.Mensagem;
```

```
    if (this.mensagem.Equals("ok")) {
```

```
        Relatorio relatorio = new Relatorio(nome, validacao.Idadei);
```

```
    }
```

```
}
```

```
public int Alternativa { get => alternativa; }
```

```
public int Numero_pergunta { get => numero_pergunta; }
```

```
internal Validacao Validacao
```

```
{
```

```
    get => default;
```

```
    set
```

```
    {
```

```
    }
```

```
}
```

```
public Frm_RegistroUsuario Frm_RegistroUsuario
```

```
{
```

```
    get => default;
```

```
    set
```

```
    {  
    }  
}  
  
public Relatorio Relatorio  
{  
    get => default;  
    set  
    {  
    }  
}  
}
```

CLASSE RELATORIO

```
using SoftwarePIM.Back_end;
using SoftwarePIM.Modelo;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Security.Cryptography.X509Certificates;
using System.Text;
using System.Threading.Tasks;

namespace SoftwarePIM.Back_End
{
    public class Relatorio : Abspropriedades
    {

        #region Atributos

        public static int total_respostas;
        public static int multi_resposta;
        public static int resultado;
        public static int soma_idade;
        public static int numero_pessoas;
        public static int total_menos18;
        public static int total_mais60;
        public static int total_entre18e60;
        public static int Idade;
        public static int media_avaliacao;
        public static string maior_reposta = "";
        public static int total_muitoruim;
        public static int total_ruim;
        public static int total_regular;
        public static int total_bom;
```

```
public static int total_muitobom;
public static int media_idade;

public List<int> lista_int_idade = new List<int>();

private string Nome = "";

#endregion

#region Constructor
public Relatorio(int mutoruim, int ruim, int regular, int bom, int muitobom)
{
    total_mutoruim += mutoruim;
    total_ruim += ruim;
    total_regular += regular;
    total_bom += bom;
    total_muitobom += muitobom;
    Relatorio.numero_pessoas++;
    calcular_media_idade();
    calcular_media();
}

public Relatorio(string nome, int Idadei)
{
    Idade = Idadei;
    Nome = nome;
}

private void calcular_media_idade()
{
```

```

    if (Idade < 18) total_menos18++;
    if (Idade > 60) total_mais60++;
    if (Idade <= 60 && Idade >= 18) total_entre18e60++;

    soma_idade = soma_idade + Idade;
    media_idade = soma_idade / numero_pessoas;

}

#endregion

#region Metodos
private void calcular_media()

{
    total_respostas = total_muitoruim + total_ruim + total_regular + total_bom +
total_muitobom;

    multi_resposta = (total_muitoruim * 1) + (total_ruim * 2) + (total_regular * 3) +
(total_bom * 4) + (total_muitobom * 5);

    resultado = multi_resposta / total_respostas;

    media_avaliacao = resultado;

    if (resultado < 2) { maior_reposta = "MUITO RUIM"; }
    else if (resultado < 3) { maior_reposta = "RUIM"; }
    else if (resultado < 4) { maior_reposta = "REGULAR"; }
    else if (resultado < 5) { maior_reposta = "BOM"; }
    else { maior_reposta = "MUITO BOM"; }

}

```

```
internal new string ToString()
{
    throw new NotImplementedException();
}
#endregion

#region propriedades

public int Total_ruim { get => total_ruim; }
public int Total_regular { get => total_regular; }
public int Total_bom { get => total_bom; }
public int Total_muitobom { get => total_muitobom; }

#endregion

public Frm_Relatorio Frm_Relatorio
{
    get => default;
    set
    {
    }
}
}
```


CLASSE VALIDAÇÃO

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;

namespace SoftwarePIM.Modelo
{
    internal class Validacao
    {
        #region Propriedades
        private int idade;
        private string mensagem;
        private string nome;
        private string idade;
        #endregion

        #region Construtor
        public Validacao(string nome, string idade)
        {
            this.nome = nome;
            this.idade = idade;
            this.Validar(nome, idade);
        }
        #endregion

        #region Metodos
        private void Validar(string nome, string idade)
        {
            this.mensagem = "ok";
        }
    }
}
```

```

int contadorEspacos = 1;
char i = '0';

try
{
    if (idade == "") { idade = "0"; }
    this.idadei = Convert.ToInt32(idade);

    foreach (char c in nome) //Verifica se esta sendo acrescido 2 espaços
entre os caracteres
    {

        if (c == ' ' && i == ' ')
        {
            contadorEspacos++;

            if (contadorEspacos >= 1)
            {
                this.mensagem = "existem 2 ou mais espaços entre os caracteres";
            }
        }
        else
        {
            contadorEspacos = 0;
        }
        i = c;
    }

    if (nome.Length > 255 || nome.Length < 3)
    {
        this.mensagem = "Digite seu nome (O nome deve conter entre 3 a 255
caracteres)";
    }
    else if (this.idadei < 1 || this.idadei > 110)

```

```

        {
            this.mensagem = "Digite sua idade (deve conter numero com valor entre
1 a 110)";
        }

    }
    catch (Exception)
    {
        this.mensagem = "Digite dados válidos";
    }

}
#endregion

#region Encapsulamento

public string Mensagem
{
    get { return this.mensagem; }
}
public int Idadei
{
    get { return this.idadei; }
}
#endregion

internal Back_end.Controle Controle
{
    get => default;
    set
    {
    }
}
}

```

}

CLASSE FORMS - REGISTROUSUARIO

```
using SoftwarePIM.Back_end;
using SoftwarePIM.Back_End;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Drawing.Text;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Xml.Serialization;

namespace SoftwarePIM
{
    public partial class Frm_RegistroUsuario : Form
    {
        private bool maiusculo = false;
        private string frase = "";
        private string frase_idade = "";
        private bool enter = false;

        public Frm_RegistroUsuario()
        {
            InitializeComponent();
            this.pnlTeclado_Nome.Hide();
            this.pnlTeclado_idade.Hide();
        }

        private void Validar(string nome, string idade)
```

```
{

    Controle controle = new Controle(nome, idade);

    if (controle.mensagem.Equals("ok"))

    {
        var frm_TelaPerguntas = new Frm_TelaPerguntas();
        frm_TelaPerguntas.FormAnterior = this;
        frm_TelaPerguntas.FormPrincipal = this.FormPrincipal;
        frm_TelaPerguntas.Show();
        frm_TelaPerguntas.BringToFront();
    }

    else

    {

        lbl_validacao.Text = controle.mensagem;

    }

}

private void Teclado(string dig)

{
    if (dig == "Enter" && this.enter == false) { this.enter = true; dig = ""; }

    if (!dig.Equals("Backspace") && this.enter == false)

    {
        if (maiusculo == false)
        {
            dig = dig.ToLower();
        }
    }
}
```

```

        txb_NomeUsuario.Text = frase += dig;
    }

    else if (dig.Equals("Backspace") && this.enter == false)
    {
        if (!string.IsNullOrEmpty(frase))
        {
            frase = frase.Remove(frase.Length - 1);

            txb_NomeUsuario.Text = frase;
        }
    }
    else if (dig.Equals("Backspace") && this.enter == true)
    {

        if (!string.IsNullOrEmpty(frase_idade))
        {
            frase_idade = frase_idade.Remove(frase_idade.Length - 1);

            txb_IdadeUsuario.Text = frase_idade;
        }
    }

    else if (this.enter == true)
    {

        if (!dig.Equals("Enter") && this.frase_idade.Count() < 3)
txb_IdadeUsuario.Text = frase_idade += dig;
    }

}

```

```
private void btn_Confirmar_Click(object sender, EventArgs e)
{
    Validar(frase, frase_idade);
}

private void btn_VoltarMenu_Click(object sender, EventArgs e)
{
    this.Close();
}

private void Frm_RegistroUsuario_FormClosed(object sender,
FormClosedEventArgs e)
{
    this.FormPrincipal?.Show();
}

private void btnCapslock_Click(object sender, EventArgs e)
{
    if (this.maiusculo == false)
    {
        this.maiusculo = true;
    }
    else
    {
        this.maiusculo = false;
    }
}

#region teclado

private void btnBackspace_Click(object sender, EventArgs e)
{
    this.Teclado("Backspace");
}
```



```
}
```

```
private void btnBackspace_Idade_Click(object sender, EventArgs e)
{
    this.Teclado("Backspace");
}
```

```
private void btn1_Click(object sender, EventArgs e)
{
    this.Teclado("1");
}
```

```
private void btn2_Click(object sender, EventArgs e)
{
    this.Teclado("2");
}
```

```
private void btn3_Click(object sender, EventArgs e)
{
    this.Teclado("3");
}
```

```
private void btn4_Click(object sender, EventArgs e)
{
    this.Teclado("4");
}
```

```
private void btn5_Click(object sender, EventArgs e)
{
    this.Teclado("5");
}
```

```
private void btn6_Click(object sender, EventArgs e)
{
```

```
        this.Teclado("6");
    }

    private void btn7_Click(object sender, EventArgs e)
    {
        this.Teclado("7");
    }

    private void btn8_Click(object sender, EventArgs e)
    {
        this.Teclado("8");
    }

    private void btn9_Click(object sender, EventArgs e)
    {
        this.Teclado("9");
    }

    private void btn0_Click(object sender, EventArgs e)
    {
        this.Teclado("0");
    }

    private void btnQ_Click(object sender, EventArgs e)
    {
        this.Teclado("Q");
    }

    private void btnW_Click(object sender, EventArgs e)
    {
        this.Teclado("W");
    }

    private void btnE_Click(object sender, EventArgs e)
```

```
{  
    this.Teclado("E");  
}
```

```
private void btnR_Click(object sender, EventArgs e)  
{  
    this.Teclado("R");  
}
```

```
private void btnT_Click(object sender, EventArgs e)  
{  
    this.Teclado("T");  
}
```

```
private void btnY_Click(object sender, EventArgs e)  
{  
    this.Teclado("Y");  
}
```

```
private void btnU_Click(object sender, EventArgs e)  
{  
    this.Teclado("U");  
}
```

```
private void btnI_Click(object sender, EventArgs e)  
{  
    this.Teclado("I");  
}
```

```
private void btnO_Click(object sender, EventArgs e)  
{  
    this.Teclado("O");  
}
```

```
private void btnP_Click(object sender, EventArgs e)
{
    this.Teclado("P");
}
```

```
private void btnA_Click(object sender, EventArgs e)
{
    this.Teclado("A");
}
```

```
private void btnS_Click(object sender, EventArgs e)
{
    this.Teclado("S");
}
```

```
private void btnD_Click(object sender, EventArgs e)
{
    this.Teclado("D");
}
```

```
private void btnF_Click(object sender, EventArgs e)
{
    this.Teclado("F");
}
```

```
private void btnG_Click(object sender, EventArgs e)
{
    this.Teclado("G");
}
```

```
private void btnH_Click(object sender, EventArgs e)
{
    this.Teclado("H");
}
```

```
private void btnJ_Click(object sender, EventArgs e)
{
    this.Teclado("J");
}
```

```
private void btnK_Click(object sender, EventArgs e)
{
    this.Teclado("K");
}
```

```
private void btnL_Click(object sender, EventArgs e)
{
    this.Teclado("L");
}
```

```
private void btnZ_Click(object sender, EventArgs e)
{
    this.Teclado("Z");
}
```

```
private void btnX_Click(object sender, EventArgs e)
{
    this.Teclado("X");
}
```

```
private void btnC_Click(object sender, EventArgs e)
{
    this.Teclado("C");
}
```

```
private void btnV_Click(object sender, EventArgs e)
{
    this.Teclado("V");
}
```

```
}
```

```
private void btnB_Click(object sender, EventArgs e)
{
    this.Teclado("B");
}
```

```
private void btnN_Click(object sender, EventArgs e)
{
    this.Teclado("N");
}
```

```
private void btnM_Click(object sender, EventArgs e)
{
    this.Teclado("M");
}
```

```
private void btnSpace_Click(object sender, EventArgs e)
{
    this.Teclado(" ");
}
```

```
#endregion
```

```
private void txb_NomeUsuario_TextChanged(object sender, EventArgs e)
{
}

}
```

```
private void txb_IdadeUsuario_TextChanged(object sender, EventArgs e)
{
}

}
```

```
private void lbl_validacao_Click(object sender, EventArgs e)
```

```
{  
  
}  
  
private void txb_NomeUsuario_Enter(object sender, EventArgs e)  
{  
    this.pnlTeclado_Nome.Show();  
    this.pnlTeclado_idade.Hide();  
    this.enter = false;  
  
}  
  
private void txb_NomeUsuario_Leave(object sender, EventArgs e)  
{  
  
}  
  
private void txb_IdadeUsuario_Enter(object sender, EventArgs e)  
{  
    this.pnlTeclado_idade.Show();  
    this.pnlTeclado_Nome.Hide();  
    this.enter = true;  
  
}  
  
private void txb_IdadeUsuario_Leave(object sender, EventArgs e)  
{  
  
}  
  
public Form? FormAnterior { get; set; }  
public Form? FormPrincipal { get; set; }  
public Form? FormAnterior2 { get; set; }
```

```
internal Controle Controle
{
    get => default;
    set
    {
    }
}

public Frm_TelaPerguntas Frm_TelaPerguntas
{
    get => default;
    set
    {
    }
}
}
```


CLASSE FORMS – RELATORIO

```
using SoftwarePIM.Back_end;
using SoftwarePIM.Back_End;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Reflection.Emit;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace SoftwarePIM
{
    public partial class Frm_Relatorio : Form

    {

        public Form? FormAnterior { get; set; }
        public Form? FormAnterior2 { get; set; }
        public Form? FormPrincipal { get; set; }

        public Frm_Relatorio()
        {
            InitializeComponent();

            lbl_QtdMuitoRuim.Text = Relatorio.total_muitoruim.ToString();
            lbl_QtdRuim.Text = Relatorio.total_ruim.ToString();
            lbl_QtdRegular.Text = Relatorio.total_regular.ToString();
        }
    }
}
```

```

        lbl_QtdBom.Text = Relatorio.total_bom.ToString();
        lbl_QtdMuitoBom.Text = Relatorio.total_muitobom.ToString();
        lbl_MediaAvaliacao.Text = Relatorio.maior_reposta;
        lbl_Qtd60.Text = Relatorio.total_mais60.ToString();
        lbl_Qtd18Menos.Text = Relatorio.total_menos18.ToString();
        lbl_Qtd18a60.Text = Relatorio.total_entre18e60.ToString();
        lbl_MediaEtaria.Text = Relatorio.media_idade.ToString();
        lbl_Total_Pessoas.Text = Relatorio.numero_pessoas.ToString();
    }

```

```

private void Frm_Relatorio_Load(object sender, EventArgs e)
{
    FormAnterior?.Close();
    FormAnterior2?.Close();
    this.FormPrincipal?.Hide();
}

```

```

private void Frm_Relatorio_FormClosing(object sender, FormClosingEventArgs
e)
{
    this.FormPrincipal?.Show();
}

```

```

private void btn_Confirmar_Click(object sender, EventArgs e)
{
    this.Close();
    this.FormPrincipal?.Show();
}

```

```
private void lbl_QtdBom_Click(object sender, EventArgs e)
{

}

private void lbl_Qtd60_Click(object sender, EventArgs e)
{

}
}
}
```

CLASSE FORMS – HISTORIA

```
using SoftwarePIM.Properties;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics.Metrics;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace SoftwarePIM
{
    public partial class Frm_TelaHistoria : Form
    {
        public Form? FormAnterior { get; set; }
        public Form? FormPrincipal { get; set; }

        private int painelAtual = 1;
        private int indiceAtual = 0;

        public Frm_TelaHistoria()
        {
            InitializeComponent();

            this.pictureBox1.Image = Resources.Obra5_3;
            timer1.Interval = 3000;
            timer1.Tick += Timer1_Tick;
            timer1.Start();
        }
    }
}
```

```
}  
private int counter = 1;  
private void Timer1_Tick(object sender, EventArgs e)  
{  
    counter++;  
    switch (counter % 3)  
    {  
        case 0:  
            this.pictureBox1.Image = Resources.Obra5_1;  
            break;  
        case 1:  
            this.pictureBox1.Image = Resources.Obra5_2;  
            break;  
        case 2:  
            this.pictureBox1.Image = Resources.Obra5_3;  
            break;  
    }  
}  
  
private void btn_Menu_Click(object sender, EventArgs e)  
{  
    this.Close();  
}  
  
private void Frm_TelaHistoria_Load(object sender, EventArgs e)  
{  
    this.FormPrincipal?.Hide();  
}  
  
private void Frm_TelaHistoria_FormClosing(object sender,  
FormClosingEventArgs e)  
{  
    this.FormPrincipal?.Show();  
}
```

```
private void btn_Proximo_Click(object sender, EventArgs e)
{
    switch (painelAtual)
    {
        case 1:
            panel1.Visible = false;
            break;
        case 2:
            panel2.Visible = false;
            break;
        case 3:
            panel3.Visible = false;
            break;
        case 4:
            panel4.Visible = false;
            break;
        case 5:
            panel5.Visible = false;
            break;
    }

    painelAtual++;
    if (painelAtual > 5)
        painelAtual = 1;

    switch (painelAtual)
    {
        case 1:
            panel1.Visible = true;
            break;
        case 2:
            panel2.Visible = true;
            break;
```

```
        case 3:
            panel3.Visible = true;
            break;
        case 4:
            panel4.Visible = true;
            break;
        case 5:
            panel5.Visible = true;
            break;
    }

}

private void btn_Anterior_Click(object sender, EventArgs e)
{
    switch (painelAtual)
    {
        case 1:
            panel1.Visible = false;
            break;
        case 2:
            panel2.Visible = false;
            break;
        case 3:
            panel3.Visible = false;
            break;
        case 4:
            panel4.Visible = false;
            break;
        case 5:
            panel5.Visible = false;
            break;
    }
}
```

```
painelAtual--;  
if (painelAtual < 1)  
    painelAtual = 5; // Altere para o número total de painéis  
  
switch (painelAtual)  
{  
    case 1:  
        panel1.Visible = true;  
        break;  
    case 2:  
        panel2.Visible = true;  
        break;  
    case 3:  
        panel3.Visible = true;  
        break;  
    case 4:  
        panel4.Visible = true;  
        break;  
    case 5:  
        panel5.Visible = true;  
        break;  
}  
}  
}
```


CLASSE FORMS – TELA INICIAL

```
namespace SoftwarePIM
{
    public partial class Frm_TelaInicial : Form
    {
        public Frm_TelaInicial()
        {
            InitializeComponent();
        }

        public Frm_TelaHistoria Frm_TelaHistoria
        {
            get => default;
            set
            {
            }
        }

        public Frm_TelaMapa Frm_TelaMapa
        {
            get => default;
            set
            {
            }
        }

        public Frm_RegistroUsuario Frm_RegistroUsuario
        {
            get => default;
            set
            {
            }
        }
    }
}
```

```
}
```

```
private void btn_Mapa_Click(object sender, EventArgs e)
{
    // Criando uma instância da Tela do Mapa
    Frm_TelaMapa frm_TelaMapa = new Frm_TelaMapa();

    // Exibindo o FormSecundario
    frm_TelaMapa.Show();

    //frm_TelaMapa.FormPrincipal = this;
}
```

```
private void btn_Historia_Click(object sender, EventArgs e)
{
    // Criando uma instância do FormSecundario
    Frm_TelaHistoria frm_TelaHistoria = new Frm_TelaHistoria();

    frm_TelaHistoria.FormPrincipal = this;

    // Exibindo o FormSecundario
    frm_TelaHistoria.Show();

}
```

```
private void btn_Avaliacao_Click(object sender, EventArgs e)
{
    // Criando uma instância do FormSecundario
    Frm_RegistroUsuario frm_RegistroUsuario = new Frm_RegistroUsuario();

    // frm_RegistroUsuario.FormPrincipal = this; nao da ruim se tirar

    // Exibindo o FormSecundario
    frm_RegistroUsuario.Show();
}
```

```
}

private void Frm_TelaInicial_Load(object sender, EventArgs e)
{

}

}

}
```

CLASSE FORMS – MAPA

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace SoftwarePIM
{
    public partial class Frm_TelaMapa : Form
    {
        public Frm_TelaMapa()
        {
            InitializeComponent();
        }

        private void btn_Menu_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

CLASSE FORMS – PERGUNTAS

```
using SoftwarePIM.Back_end;
using SoftwarePIM.Back_End;
using SoftwarePIM.Modelo;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace SoftwarePIM
{
    public partial class Frm_TelaPerguntas : Form

    {
        private int alternativa;
        private int numero_pergunta;
        public Form? FormAnterior { get; set; }
        public Form? FormPrincipal { get; set; }

        public Frm_TelaPerguntas()
        {
            InitializeComponent();
            Esconder();
        }
        private void btn_VoltarMenu_Click(object sender, EventArgs e)
        {
```

```

        this.Close();
    }

```

```

private int muito_ruim;
private int ruim;
private int regular;
private int bom;
private int muito_bom;
private int painelAtual = 1; // Variável para rastrear qual painel está atualmente
visível

```

```

private void Esconder() {

    pcb_foguetinho1Pg1.Hide();
    pcb_foguetinho2Pg1.Hide();
    pcb_foguetinho3Pg1.Hide();
    pcb_foguetinho4Pg1.Hide();
    pcb_foguetinho5Pg1.Hide();
    pcb_foguetinho1Pg2.Hide();
    pcb_foguetinho2Pg2.Hide();
    pcb_foguetinho3Pg2.Hide();
    pcb_foguetinho4Pg2.Hide();
    pcb_foguetinho5Pg2.Hide();
    pcb_foguetinho1Pg3.Hide();
    pcb_foguetinho2Pg3.Hide();
    pcb_foguetinho3Pg3.Hide();
    pcb_foguetinho4Pg3.Hide();
    pcb_foguetinho5Pg3.Hide();
    pcb_foguetinho1Pg4.Hide();
    pcb_foguetinho2Pg4.Hide();
    pcb_foguetinho3Pg4.Hide();
    pcb_foguetinho4Pg4.Hide();
    pcb_foguetinho5Pg4.Hide();
    pcb_foguetinho1Pg5.Hide();

```

```

        pcb_foguettino2Pg5.Hide();
        pcb_foguettino3Pg5.Hide();
        pcb_foguettino4Pg5.Hide();
        pcb_foguettino5Pg5.Hide();

    }

    private void mandar()

    {
        Controle controle = new
        Controle(muito_ruim,ruim,regular,bom,muito_bom,numero_pergunta);

    }

    private void btn_ProximaPergunta_Click(object sender, EventArgs e)

    {
        if (alternativa == 1) muito_ruim++;
        if (alternativa == 2) ruim++;
        if (alternativa == 3) regular++;
        if (alternativa == 4) bom++;
        if (alternativa == 5) muito_bom++;
        if (numero_pergunta == 5) mandar();

        // Esconde o painel atual
        switch (painelAtual)
        {
            case 1:
                panel1.Visible = false;
                // this.numero_pergunta = 0;
                break;
            case 2:
                panel2.Visible = false;

```

```

        // this.numero_pergunta = 0;
        break;
    case 3:
        panel3.Visible = false;
        // this.numero_pergunta = 0;
        break;
    case 4:
        panel4.Visible = false;
        // this.numero_pergunta = 0;
        break;
    case 5:
        panel5.Visible = false;
        // this.numero_pergunta = 0;
        break;
    }

    if (alternativa != 0 )
    {
        painelAtual++;
    }

    if (painelAtual > 5)
    {

        painelAtual = 1;
        // Abre um novo formulário
        AbrirNovoFormulario();
    }
    else
    {

        // Exibe o próximo painel
        switch (painelAtual)
        {
            case 1:

```



```

        panel1.Visible = true;

        break;
    case 2:
        panel2.Visible = true;

        break;
    case 3:
        panel3.Visible = true;

        break;
    case 4:
        panel4.Visible = true;

        break;
    case 5:
        panel5.Visible = true;

        break;
    }
}

alternativa = 0;

}

private void AbrirNovoFormulario()

{
    var frm_Relatorio = new Frm_Relatorio();
    frm_Relatorio.FormAnterior = this;
    frm_Relatorio.FormPrincipal = this.FormPrincipal;
    frm_Relatorio.Show();
    frm_Relatorio.BringToFront();
}

```

```
private void Frm_Pergunta1_Load(object sender, EventArgs e)
{
    FormAnterior?.Close();
    this.FormPrincipal?.Hide();
}

private void Frm_Pergunta1_FormClosing(object sender,
FormClosingEventArgs e)
{
    this.FormPrincipal?.Show();
}

private void pcb_BomPg1_Click(object sender, EventArgs e)
{
    Esconder();
    pcb_foguettino4Pg1.Show();
    this.alternativa = 4;
    this.numero_pergunta = 1;
}

private void pcb_BomPg2_Click(object sender, EventArgs e)
{
    Esconder();
    pcb_foguettino4Pg2.Show();
    this.alternativa = 4;
    this.numero_pergunta = 2;
}

private void pcb_BomPg3_Click(object sender, EventArgs e)
{
    Esconder();
```

```
    pcb_foguettino4Pg3.Show();  
    this.alternativa = 4;  
    this.numero_pergunta = 3;  
}
```

```
private void pcb_BomPg4_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguettino4Pg4.Show();  
    this.alternativa = 4;  
    this.numero_pergunta = 4;  
}
```

```
private void pcb_BomPg5_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguettino4Pg5.Show();  
    this.alternativa = 4;  
    this.numero_pergunta = 5;  
}
```

```
private void pcb_MuitoBomPg1_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguettino5Pg1.Show();  
    this.alternativa = 5;  
    this.numero_pergunta = 1;  
}
```

```
private void pcb_MuitoBomPg2_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguettino5Pg2.Show();  
    this.alternativa = 5;
```

```
        this.numero_pergunta = 2;
    }

    private void pcb_MuitoBomPg3_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguetinho5Pg3.Show();
        this.alternativa = 5;
        this.numero_pergunta = 3;
    }

    private void pcb_MuitoBomPg4_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguetinho5Pg4.Show();
        this.alternativa = 5;
        this.numero_pergunta = 4;
    }

    private void pcb_MuitoBomPg5_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguetinho5Pg5.Show();
        this.alternativa = 5;
        this.numero_pergunta = 5;
    }

    private void pcb_MuitoRuimPg1_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguetinho1Pg1.Show();
        this.alternativa = 1;
        this.numero_pergunta = 1;
    }
}
```

```
private void pcb_MuitoRuimPg2_Click(object sender, EventArgs e)
{
    Esconder();
    pcb_foguettino1Pg2.Show();
    this.alternativa = 1;
    this.numero_pergunta = 2;
}
```

```
private void pcb_MuitoRuimPg3_Click(object sender, EventArgs e)
{
    Esconder();
    pcb_foguettino1Pg3.Show();
    this.alternativa = 1;
    this.numero_pergunta = 3;
}
```

```
private void pcb_MuitoRuimPg4_Click(object sender, EventArgs e)
{
    Esconder();
    pcb_foguettino1Pg4.Show();
    this.alternativa = 1;
    this.numero_pergunta = 4;
}
```

```
private void pcb_MuitoRuimPg5_Click(object sender, EventArgs e)
{
    Esconder();
    pcb_foguettino1Pg5.Show();
    this.alternativa = 1;
    this.numero_pergunta = 5;
}
```

```
private void pcb_RegularPg1_Click(object sender, EventArgs e)
```

```
{  
    Esconder();  
    pcb_foguetinho3Pg1.Show();  
    this.alternativa = 3;  
    this.numero_pergunta = 1;  
}
```

```
private void pcb_RegularPg2_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguetinho3Pg2.Show();  
    this.alternativa = 3;  
    this.numero_pergunta = 2;  
}
```

```
private void pcb_RegularPg3_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguetinho3Pg3.Show();  
    this.alternativa = 3;  
    this.numero_pergunta = 3;  
}
```

```
private void pcb_RegularPg4_Click(object sender, EventArgs e)  
{  
    Esconder();  
    pcb_foguetinho3Pg4.Show();  
    this.alternativa = 3;  
    this.numero_pergunta = 4;  
}
```

```
private void pcb_RegularPg5_Click(object sender, EventArgs e)  
{  
    Esconder();
```

```
        pcb_foguettino3Pg5.Show();
        this.alternativa = 3;
        this.numero_pergunta = 5;
    }

    private void pcb_RuimPg1_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguettino2Pg1.Show();
        this.alternativa = 2;
        this.numero_pergunta = 1;
    }

    private void pcb_RuimPg2_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguettino2Pg2.Show();
        this.alternativa = 2;
        this.numero_pergunta = 2;
    }

    private void pcb_RuimPg3_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguettino2Pg3.Show();
        this.alternativa = 2;
        this.numero_pergunta = 3;
    }

    private void pcb_RuimPg4_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguettino2Pg4.Show();
        this.alternativa = 2;
```

```
        this.numero_pergunta = 4;
    }

    private void pcb_RuimPg5_Click(object sender, EventArgs e)
    {
        Esconder();
        pcb_foguetinho2Pg5.Show();
        this.alternativa = 2;
        this.numero_pergunta = 5;
    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    internal Controle Controle
    {
        get => default;
        set
        {
        }
    }
}
}
```


APÊNDICE B – MANUAL DO USUÁRIO

SOFTWARE PARA TOTEM DE INTEREÇÃO E AVALIAÇÃO



SOROCABA

2024

SOFTWARE DO TOTEM

Seja bem-vindo ao Software do Totem! Este manual fornecerá todas as informações necessárias para utilizar todas as funcionalidades deste software de maneira eficaz e eficiente.

INTRODUÇÃO

O Software do Totem é uma chave poderosa projetada para criar e gerenciar conteúdos interativos em totens digitais. Ele oferece uma interface intuitiva para criar experiências personalizadas, fornecer informações úteis e engajar os usuários de maneira eficaz.

INTERFACE DO USUÁRIO:

A interface do usuário do Software do Totem é composta pelos seguintes elementos principais:

Tela principal: Onde os usuários podem acessar todas as funcionalidades do software, incluindo Mapa, Histórias e Avaliação.

Tela Mapa: Onde os usuários podem acessar todo o mapa do museu e localização das obras expostas.

Tela História: Permite aos usuários visualizar os textos descritivos sobre a exposição de cada obra do museu.

Tela Avaliação: Permite aos usuários avaliar as obras expostas no museu com a visualização de relatórios.

INSTRUÇÕES DE USO:

- **DADO** que o visitante deseja saber mais sobre o museu
- **E** deixar sua avaliação sobre a exposição
- **QUANDO** o visitante se dirigir-se ao totem
- **ENTÃO** deverá ver a tela de menu com as opções: “MAPA, HISTÓRIA e AVALIAÇÃO
- **E** a mensagem no rodapé “Nós respeitamos sua privacidade e seguimos as normas da LGPD, mesmo tratando apenas de dados não pessoais. Sua proteção é nossa prioridade.”



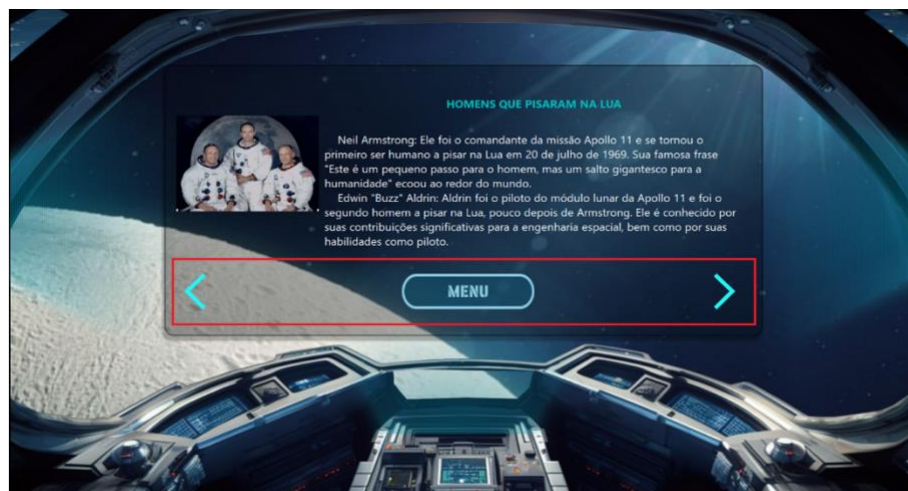
MAPA

- **DADO** que o visitante esteja no menu do totem
- **QUANDO** selecionar o botão “MAPA”
- **ENTÃO** o totem deverá apresentar o mapa do museu
- **E** o botão “MENU” para voltar ao Menu.



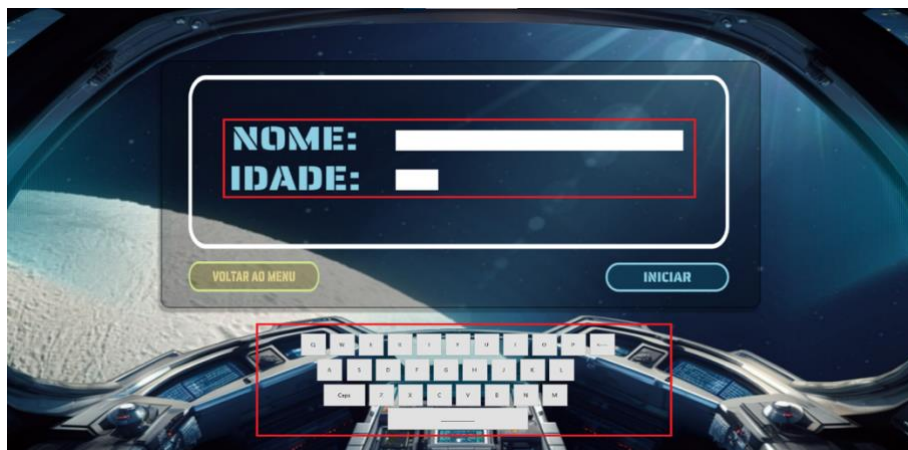
HISTÓRIA:

- **DADO** que o visitante esteja no menu do totem;
- **QUANDO** selecionar o botão “HISTÓRIA”;
- **ENTÃO** o totem deverá apresentar a tela da “Obras” com suas informações e textos descritivos sobre a exposição;
- **E** os botões “<” Voltar, “MENU” e “>” Avançar.



AVALIAÇÃO:

- **DADO** que o visitante esteja no menu do totem;
- **QUANDO** selecionar o botão “AVALIAÇÃO”;
- **ENTÃO** o totem deverá apresentar a tela de preenchimento de “NOME” e “IDADE”;
- **E** os botões “VOLTAR AO MENU” e “INICIAR”;
- **E** o teclado virtual.

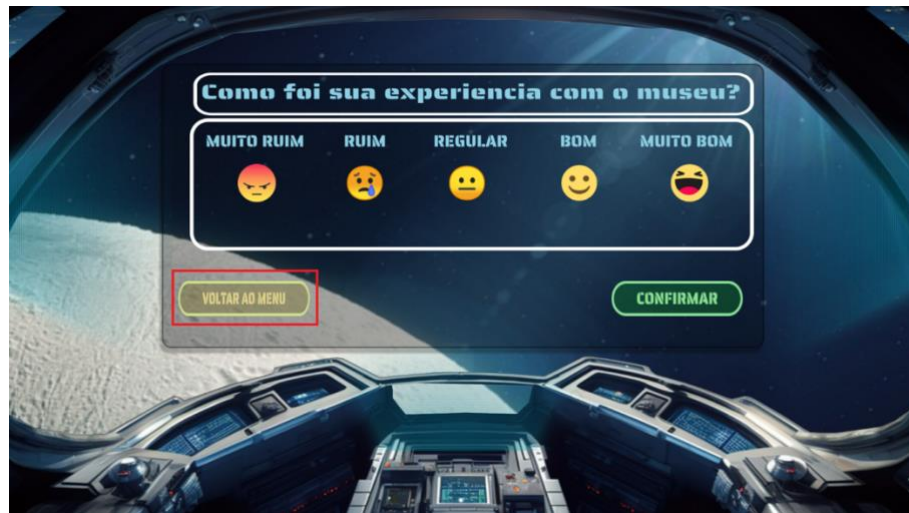




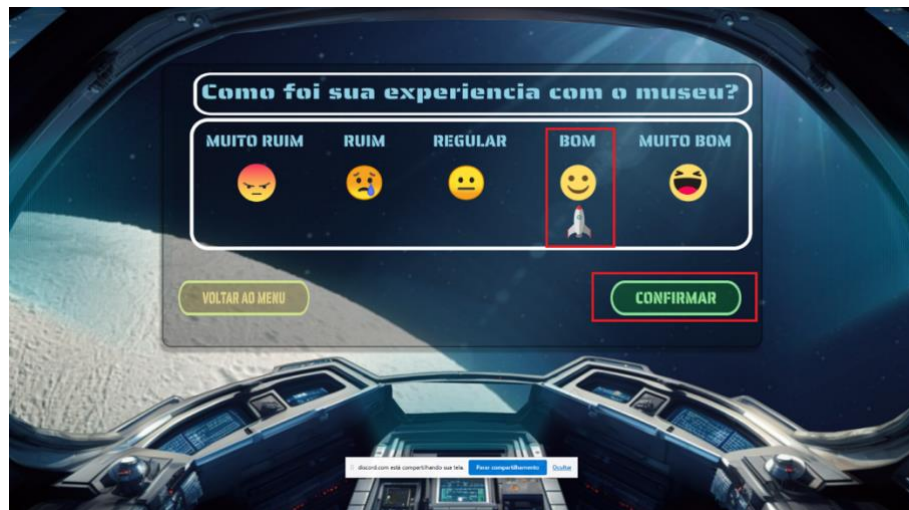
- **DADO** que o visitante esteja na tela de preenchimento de “NOME” e “IDADE”;
- **E** preencher os campos “NOME” e “IDADE” válidos;
- **QUANDO** selecionar o botão “INICIAR”;
- **ENTÃO** o totem deverá apresentar a tela da “Pergunta 1”;
- **E** os botões “VOTAR AO MENU” e “CONFIRMAR”.



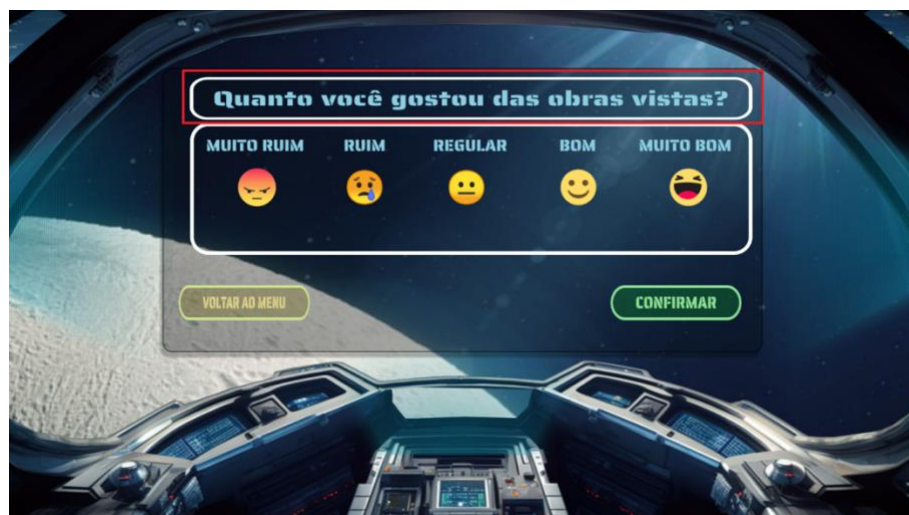
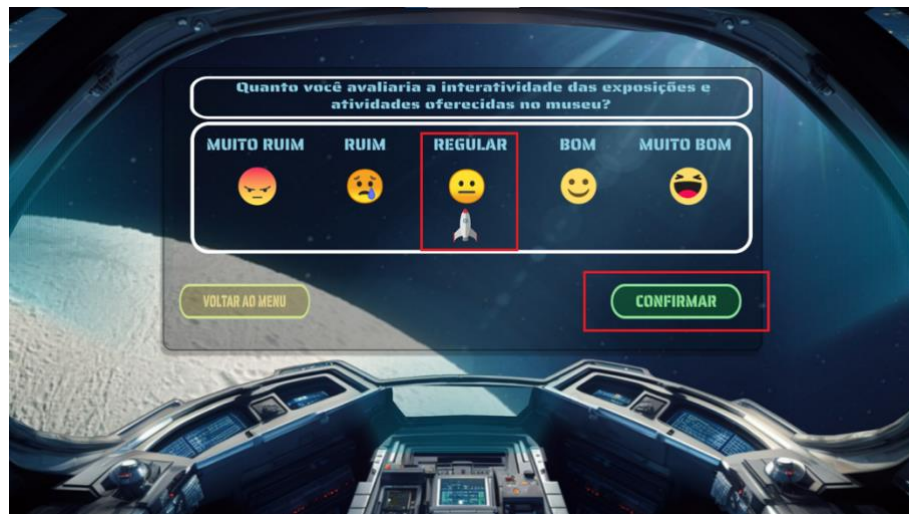
- **DADO** que o visitante esteja na tela da “Pergunta 1”;
- **QUANDO** selecionar o botão “VOLTAR AO MENU”;
- **ENTÃO** o visitante deverá ser direcionado a tela de menu com as opções: “MAPA, HISTÓRIA e AVALIAÇÃO”.



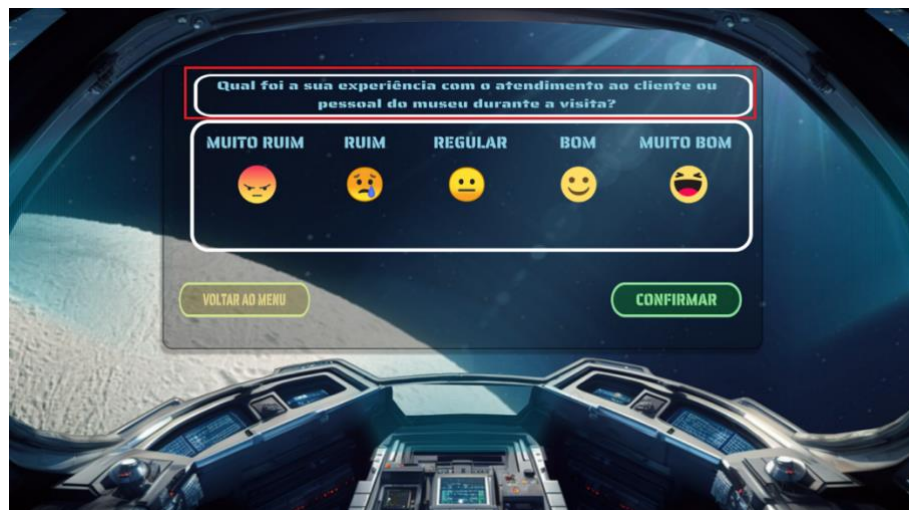
- **DADO** que o visitante esteja na tela da primeira “Pergunta 1”
- **E** escolher uma alternativa
- **QUANDO** selecionar o botão “CONFIRMAR”
- **ENTÃO** o totem deverá apresentar a tela da “Pergunta 2”
- **E** os botões “VOTAR AO MENU” e “CONFIRMAR”.



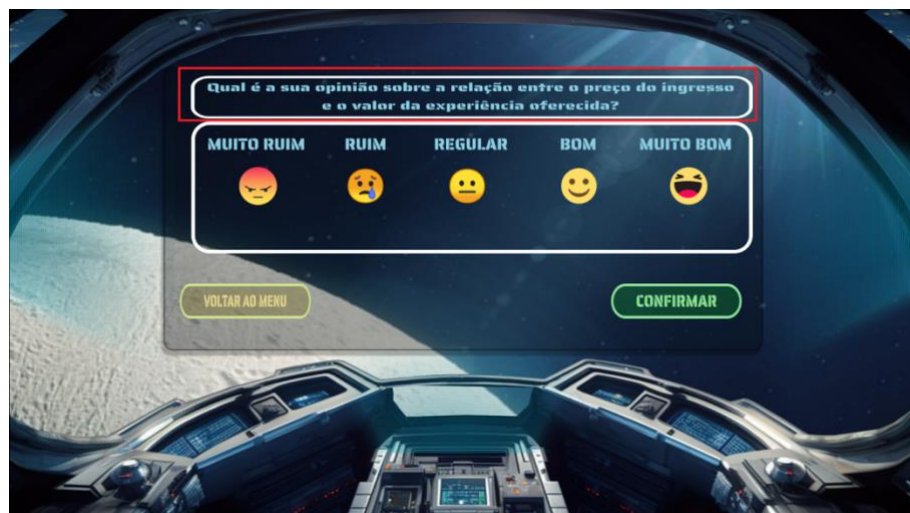
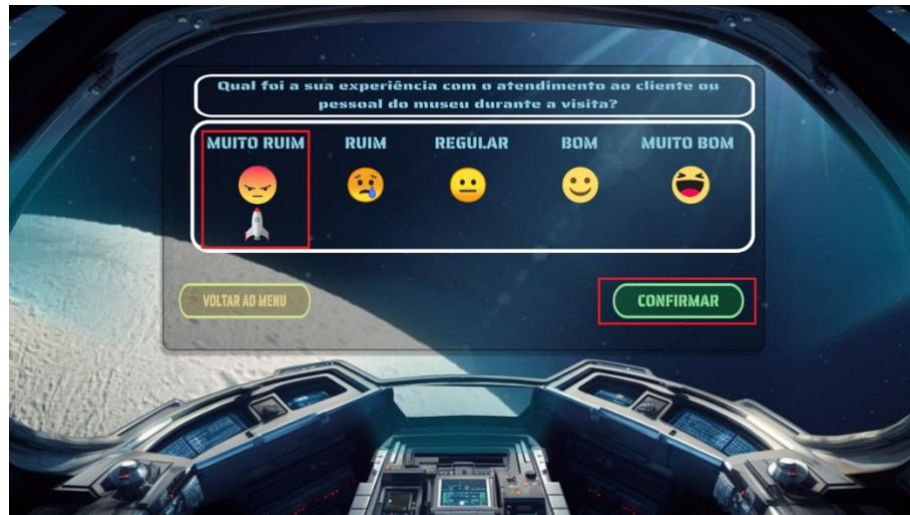
- **DADO** que o visitante esteja na tela da primeira “Pergunta 2”
- **E** escolher uma alternativa
- **QUANDO** selecionar o botão “CONFIRMAR”
- **ENTÃO** o totem deverá apresentar a tela da “Pergunta 3”
- **E** os botões “VOTAR AO MENU” e “CONFIRMAR”.



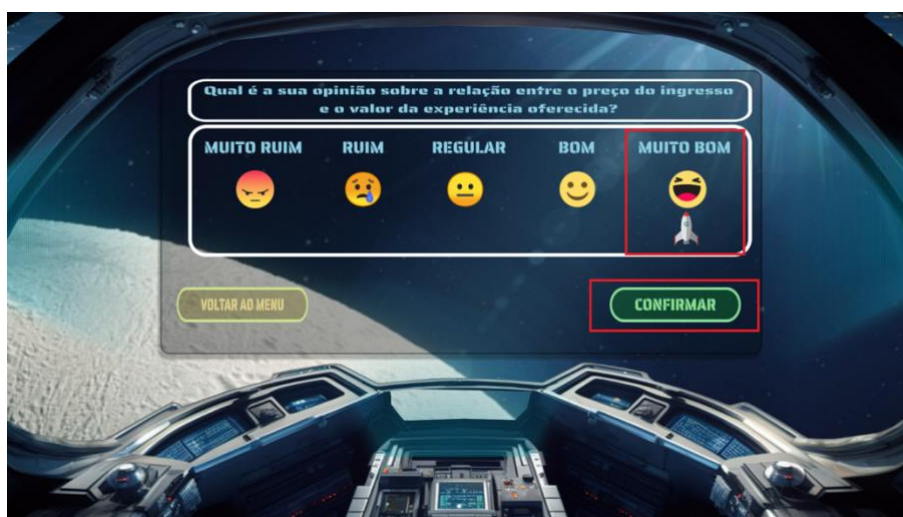
- **DADO** que o visitante esteja na tela da primeira “Pergunta 3”
- **E** escolher uma alternativa
- **QUANDO** selecionar o botão “CONFIRMAR”
- **ENTÃO** o totem deverá apresentar a tela da “Pergunta 4”
- **E** os botões “VOTAR AO MENU” e “CONFIRMAR”.



- **DADO** que o visitante esteja na tela da primeira “Pergunta 4”
- **E** escolher uma alternativa
- **QUANDO** selecionar o botão “CONFIRMAR”
- **ENTÃO** o totem deverá apresentar a tela da “Pergunta 5”
- **E** os botões “VOTAR AO MENU” e “CONFIRMAR”.



- **DADO** que o visitante esteja na tela da primeira “Pergunta 5”
- **E** escolher uma alternativa
- **QUANDO** selecionar o botão “CONFIRMAR”
- **ENTÃO** o totem deverá apresentar a tela de “ANDAMENTO DAS OBRAS”
- **E** o botão “VOTAR AO MENU”



- **DADO** que o visitante esteja na tela de “ANDAMENTO DAS OBRAS”
- **QUANDO** selecionar o botão “VOLTAR AO MENU”;
- **ENTÃO** deverá retornar a tela inicial de menu com as opções: “MAPA, HISTÓRIA e AVALIAÇÃO”.



SEGURANÇA

Proteção de Dados: Mantem nome e idade seguros e protege os dados dos usuários armazenados no totem.

Acesso Restrito: Limite o acesso ao software apenas a usuários autorizados para evitar alterações não autorizadas no conteúdo do totem.

CONTATE-NOS

Se você tiver dúvidas, algum tipo de problema ou sugestão relacionada ao uso do Totem, entre em contato conosco pelo nosso suporte técnico fornecido a empresa.

Esperamos que este manual seja útil para aproveitar ao máximo o Software de Totem Interativo. Aproveite a experiência interativas envolventes para seus usuários!

APÊNDICE C – CRONOGRAMA DE GRANTT

Legenda	
OK	
Adiado	
PRAZO	
CANCELADO	

TAREFAS/DIA	14/abr	21/abr	28/abr	05/mai	12/mai	19/mai	24/mai	OBSERVAÇÃO
PROTOTIPAR AS TELAS								UTILIZAR A FERRAMENTA FIGMA
INTRODUÇÃO								LER E INTERPRETAR O ESCOPO DO PIM
RESUMO								LER E INTERPRETAR O ESCOPO DO PIM
OBJETIVOS								LER E INTERPRETAR O ESCOPO DO PIM
REQUISITOS FUNCIONAIS								USAR COMO BASE MATERIA DE ENGENHARIA DE SOFTWARE E AOO
REQUISITOS NÃO FUNCIONAIS								USAR COMO BASE MATERIA DE ENGENHARIA DE SOFTWARE E AOO
REQUISITOS DE NEGOCIOS								USAR COMO BASE MATERIA DE ENGENHARIA DE SOFTWARE E AOO
Bibliografias Gerais								NÃO UTILIZAR BLOGS COMO FONTE
DIAGRAMA DE SEQUENCIA								UTILIZAR A MATERIA DE ANALISE DE SISTEMA ORIENTADA A OBJETO
DIAGRAMA DAS CLASSES METODOS E OBJETOS								UTILIZAR AS MATERIAS ANALISE ORIENTADA A OBJETOS COM PROGRAMAÇÃO ORIENTADA A OBJETO
SOFTWARE BASE PRONTO								O SOFTWARE COM OS REQUISITOS DEFINIDOS NO TRABALHO
MANUAL DO USUARIO								MANUAL SIMPLES E CONCISO COM O MODO DE UTILIZAÇÃO DO SOFTWARE
ATUALIZAÇÕES								SE CASO TIVER NECESSIDADE DE ACRESCENTAR ALGO NO SOFTWARE OU NA DOCUMENTAÇÃO
FASE DE TESTES								FASE ONDE FAZEMOS OS TESTES NO SOFTWARE PARA QUE NÃO NOS PEGUE DE SURPRESA NA APRESENTAÇÃO
CONCLUSÃO								CONCLUSÃO DA DOCUMENTAÇÃO E DO SOFTWARE