

WAD

WEB APPLICATION DOCUMENT

InPuT

Erik Batista da Silva

Gabrielle Dias Cartaxo

Gustavo Wagon Widman

Luiza Rodrigues Santana

Murilo de Souza Prianti Silva

Pedro Faria Santos

23 de junho de 2023

Controle do Documento

Histórico de revisões

Data	Autor	Versão	Resumo da atividade
28/04/2023	Erik Silva, Gustavo Widman, Gabrielle Cartaxo, Luiza Santana, Murilo Prianti e Pedro Faria	1.0	Criação do documento e alteração dos tópicos 1, 2 e 4.
12/05/2023	Erik Silva, Gustavo Widman, Gabrielle Cartaxo, Luiza Santana, Murilo Prianti e Pedro Faria	2.0	Edição do apêndice e dos tópicos 4 e 6, além da alteração de alguns tópicos anteriores.
06/06/2023	Erik Silva, Gustavo Widman, Gabrielle Cartaxo, Luiza Santana, Murilo Prianti e Pedro Faria	3.0	Documentação dos testes de usabilidade e edição do tópico 6.1.
22/06/2023	Erik Silva, Gustavo Widman, Gabrielle Cartaxo, Luiza Santana, Murilo Prianti e Pedro Faria	4.0	Finalização da documentação.

Sumário

[Visão Geral do Projeto](#)

[Parceiro de Negócios](#)

[O Problema](#)

[Objetivos](#)

[Objetivos gerais](#)

[Objetivos específicos](#)

[Descritivo da Solução](#)

[Partes Interessadas](#)

[Análise do Problema](#)

[Análise da Indústria](#)

[Análise do cenário: Matriz SWOT](#)

[Proposta de Valor: Value Proposition Canvas](#)

[Matriz de Risco](#)

[Requisitos do Sistema](#)

[Persona](#)

[Histórias dos usuários \(user stories\)](#)

[Arquitetura do Sistema](#)

[Módulos do Sistema e Visão Geral \(Big Picture\)](#)

[Tecnologias Utilizadas](#)

[UX e UI Design](#)

[Wireframe](#)

[Design de Interface - Guia de Estilos](#)

[Projeto de Banco de Dados](#)

[Modelo Conceitual](#)

[Modelo Lógico](#)

[Testes de Software](#)

[Teste de Usabilidade](#)

[Referências](#)

[Apêndice](#)

1. Visão Geral do Projeto

1.1. Parceiro de Negócios

Com sede em São Paulo, o IPT (instituto de pesquisas e tecnologias) é uma instituição de grande porte, contando com mais de 1000 funcionários e equipamentos altamente qualificados e capazes de desenvolver alta tecnologia para desenvolver projetos. A empresa atua em diversas áreas, incluindo construção civil, energia, meio ambiente, metrologia, materiais, tecnologia da informação e ferrovias.

O IPT possui uma ampla área de atuação no mercado, estando no setor público e privado, desenvolvendo projetos em parceria com o governo, empresas de diversos segmentos, universidades e institutos de pesquisa, tanto no âmbito nacional quanto internacional.

A empresa é reconhecida como uma das principais instituições de pesquisa do Brasil por sua expertise em inovação, tendo uma posição no mercado internacional e nacional muito importante e privilegiada, em razão de seus laboratórios e pesquisadores altamente qualificados e a escassez de concorrentes em sua área de atuação.

1.2. O Problema

Atualmente, o IPT apresenta muitas vantagens em relação à concorrência, conforme retratado no livro "A estratégia do Oceano Azul" escrito por W. Chan Kim Renée Mauborgne e W. Chan Kim. Dentro do livro, podemos notar que no processo de criar um oceano azul mostra que a concorrência está se transformando em uma mera rivalidade, levando os stakeholders a buscar o IPT como a opção mais adequada e responsável para entregar um documento de cunho técnico com o mais alto nível de excelência possível. Isso envolve atender a todas as exigências e critérios impostos pelo cliente.

Com o passar do tempo, o IPT tornou-se uma instituição de renome no mercado onde atua (completa 124 anos de existência em 2023). Ao longo dos anos e com os trabalhos desenvolvidos no papel, o IPT criou diversos relatórios, o que tem funcionado para eles. Porém, os clientes estão se tornando cada vez mais exigentes e desejam relatórios de formas mais tecnológicas, a ponto de não precisarem mais percorrer uma pesquisa de 390 páginas para encontrar o que procuram no sumário e navegar até a página correspondente.

Foi nesse ponto que o IPT se deparou com o problema da acessibilidade dos relatórios via WEB, bem como da apresentação visual das informações para tornar os relatórios mais didáticos e intuitivos, permitindo que os clientes tomem as melhores decisões e encontrem os resultados desejados com apenas alguns cliques dentro do relatório.

1.3. Objetivos

1.3.1. Objetivos gerais

O projeto tem como principal objetivo o desenvolvimento de uma aplicação web que visa facilitar a organização dos relatórios produzidos pelo IPT referentes a longarina, engate e instrumentação de locomotivas. Além disso, durante a construção do produto, os desenvolvedores serão capacitados em desenvolvimento web, incluindo linguagens como HTML, JavaScript, CSS, Node e SQL.

1.3.2. Objetivos específicos

Os desenvolvedores definiram como objetivos específicos o desenvolvimento de uma plataforma que consegue, a partir da análise dos dados disponibilizados pelo IPT, apresentá-los de forma clara e intuitiva. Serão desenvolvidas ferramentas de filtragem de informação, mapas de visualização dinâmica e sistemas de busca específicos. Tudo isso será apresentado ao cliente de forma limpa e suave, numa plataforma profissional e bem desenvolvida.

1.4. Descritivo da Solução

A solução proposta é a construção de um site que entregue as informações dos relatórios de maneira mais simplificada, mas que possa ser aprofundada conforme a necessidade do usuário. As principais funcionalidades serão o acesso a diferentes dados dos trens e seus trajetos, podendo clicar nos pontos mais críticos para acessar as informações correlacionadas, tornando o conhecimento mais visual e interativo. Podendo ser acessado de qualquer navegador web, torna-se mais acessível para as partes interessadas, que serão apresentadas no tópico 1.5.

1.5. Partes Interessadas

Os principais stakeholders envolvidos no projeto são IPT, o Inteli e o grupo 1 do módulo 2. O papel do IPT é fornecer informações precisas para a obtenção de melhores resultados, outra obrigação do cliente é validar as ideias que estão sendo discutidas pelo grupo responsável pelo projeto.

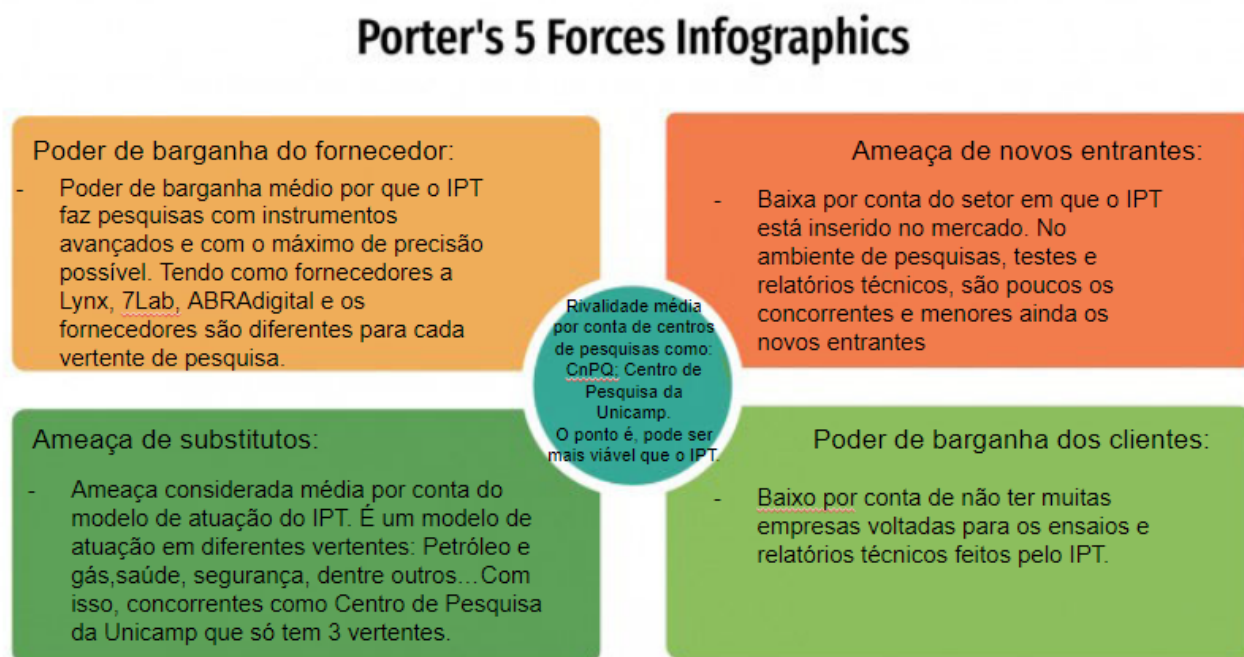
O IPT como cliente tem suas expectativas e exigências, o trabalho do grupo 1 é atender a essas expectativas e preferencialmente superá-las. O papel do Inteli já está em vigor, pois a instituição já estabeleceu a conexão entre o instituto (IPT) e os alunos do Inteli.

2. Análise do Problema

2.1. Análise da Indústria

O infográfico demonstrado abaixo visa representar as 5 forças de Porter. As 5 forças são poderes de barganha do fornecedor, ameaça de substitutos, ameaça de novos entrantes, poder de barganha de clientes e a rivalidade dos concorrentes.

Figura 1

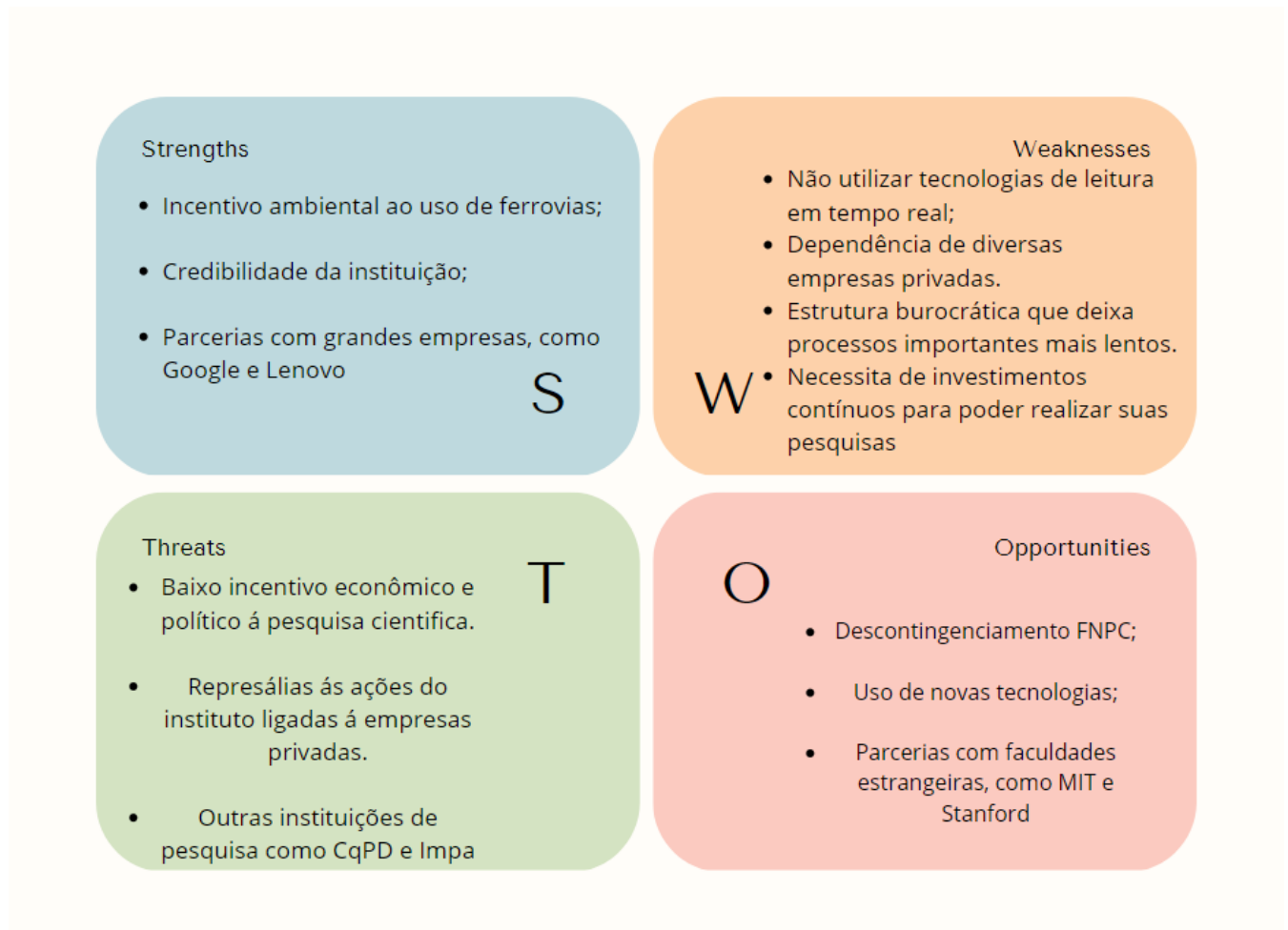


Fonte: Elaborado pelos autores.

2.2. Análise do cenário: Matriz SWOT

O esquema representado abaixo é uma análise SWOT (Figura 2). A análise SWOT demonstra as forças, fraquezas, ameaças e oportunidades do projeto.

Figura 2

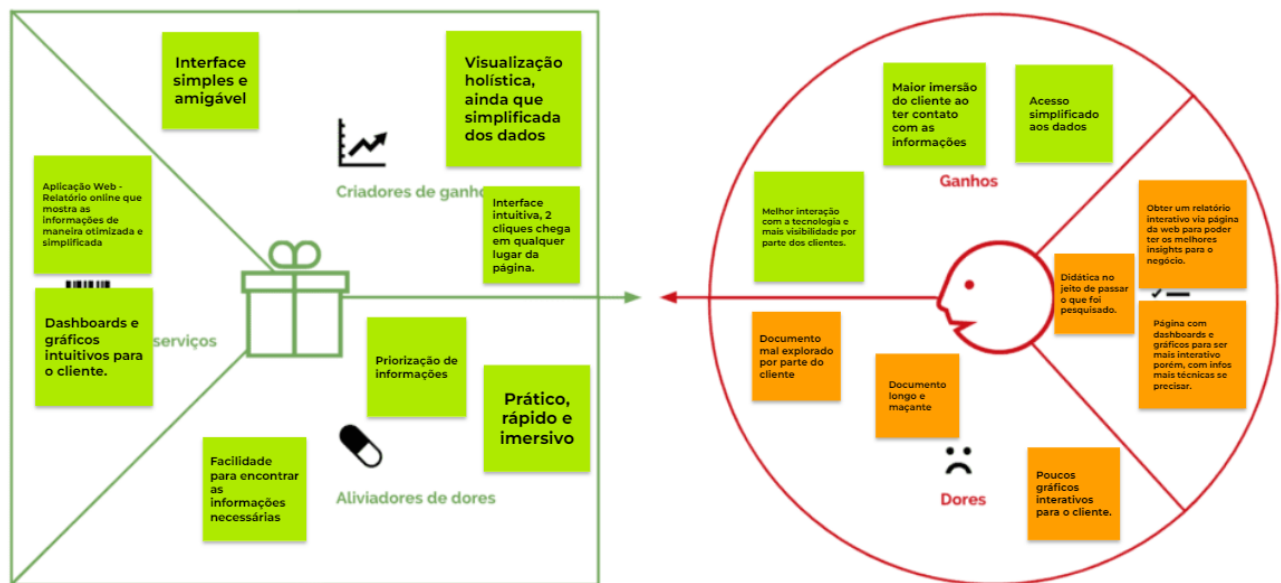


Fonte: Elaborado pelos autores

2.3. Proposta de Valor: Value Proposition Canvas

O value proposition canvas é uma ferramenta de análise de público consumidor. Tem como objetivo entender as necessidades, dores e desejos.

Figura 3



Fonte: Elaborado pelos autores.

2.4. Matriz de Risco

A matriz de risco é um gráfico que identifica e determina o tamanho de um risco, possibilitando o impedimento ou controle de contratempos.

Figura 4

Probabilidade	Ameaças					Oportunidades				
	90%			Muito conteúdo para um só módulo			Aprender JS, CSS, HTML.			
	70%		Mal entendimento do projeto		Página não agradar o cliente	Erro nos botões na pág do relatório	entendimento para futuros projetos	Entregar mais que o esperado	Aprendizado em banco de dados (SQLite)	
	50%		Escopo do projeto mal feito	Não entregar o projeto a tempo	Integração dos mapas na página WEB	Não cumprir as promessas para o projeto.		aprender sobre API de sites		
	30%		Intrigas na equipe			Atividade acumulada entre as sprints				implementação da nossa página no site do IPT
	10%			Dar erro na página da WEB						
		Muito Baixo	Baixo	Moderado	Alto	Muito Alto	Muito Alto	Alto	Moderado	Baixo
		Impacto								

Fonte: Elaborado pelos autores.

Esta tabela (Tabela 1) foi feita mediante uma atividade onde discutimos sobre possíveis ameaças e oportunidades do projeto, assim passando todas essas informações discutidas para a tabela, onde ela traz medidas e responsáveis para esses possíveis acontecimentos.

Tabela 1 (Fonte: Elaborado pelos autores).

Ameaças e Oportunidades	Ação	Responsável (Mediador)	Medida
Atividade acumulada entre as sprints	Mitigar	Toda a equipe	Redistribuir as atividades priorizando as de maior importância.
Intrigas na equipe	Mitigar	Scrum Master da semana	Entender os motivos da equipe e reunir todos para conversar sobre o problema encontrado.
Mal entendimento do projeto	Eliminar	Toda a equipe	Re-estudar o projeto para criar um novo escopo considerando o tempo restante para a entrega do projeto.
Não entregar o projeto a tempo	Eliminar	Scrum Master da semana	Através da definição de pronto do projeto, avaliar a cada sprint o

			estado da aplicação para mudar as metas das sprints.
Escopo do projeto mal feito	Eliminar	Toda a equipe	Estudar melhor o projeto e montar um escopo que siga melhor a meta final.
Muito conteúdo para um só módulo	Mitigar	Scrum Master da Semana	Fazer as dailys de forma eficiente para entender as necessidades de cada um, evitando atrasos nos conteúdos.
Página não agradar o cliente	Eliminar	Toda a equipe	Perguntar ao cliente quais as mudanças necessárias para atingir as expectativas.
Dar erro na página WEB	Eliminar	Devs da semana	Debugar o código da página
Aprender JS, CSS e HTML	Aprimorar	Toda a equipe	Evoluir e aprender cada um dos temas mais profundamente.
Entregar mais que o esperado	Aprimorar	Toda a equipe	Para isso deve-se concluir tudo esperado como mínimo para o entregável.

Requisitos do Sistema

2.5. Persona

Personas são representações fictícias de usuários que ajudam na definição de um público alvo, assim podendo definir melhor o que o cliente procura e quais são suas dores, desejos e motivações . Elas

foram feitas mediante uma conversa em grupo onde descrevemos e pensamos nos usuários que utilizariam nossa aplicação WEB. Conforme apresentado nas figuras (5, 6 e 7)

Figura 5 - Persona 1

César



57 anos
Engenheiro Mecânico
Unicamp

BIO

Engajado na área de ferrovias, gosta muito de passar tempo com a família e mora em Brasília

Necessidades

- Ter melhor relações com seu companheiro de trabalho
- Ter mais tempo com sua família
- Maior prática com a tecnologia para aprimorar suas ações no trabalho

Frustrações

- Não tem a paciência de aprender mais sobre tecnologia
- Extremamente impaciente com erros de colegas de trabalho
- Lida muito mal com mudanças na área de trabalho

"

- Super interessado na sua área de trabalho e adora saber como os objetos funcionam

Fonte: Elaborado pelos autores.

Figura 6 - Persona 2

Susana



54 anos
Executiva, Doutorado
em finanças
corporativas na USP

BIO

Moradora de Higienópolis São Paulo, Executiva do IPT e adora trabalhar

Necessidades

- Construir uma boa carreira profissional
- Realizar atividades físicas para sair do trabalho
- Relatórios mais práticos para os seus clientes

Frustrações

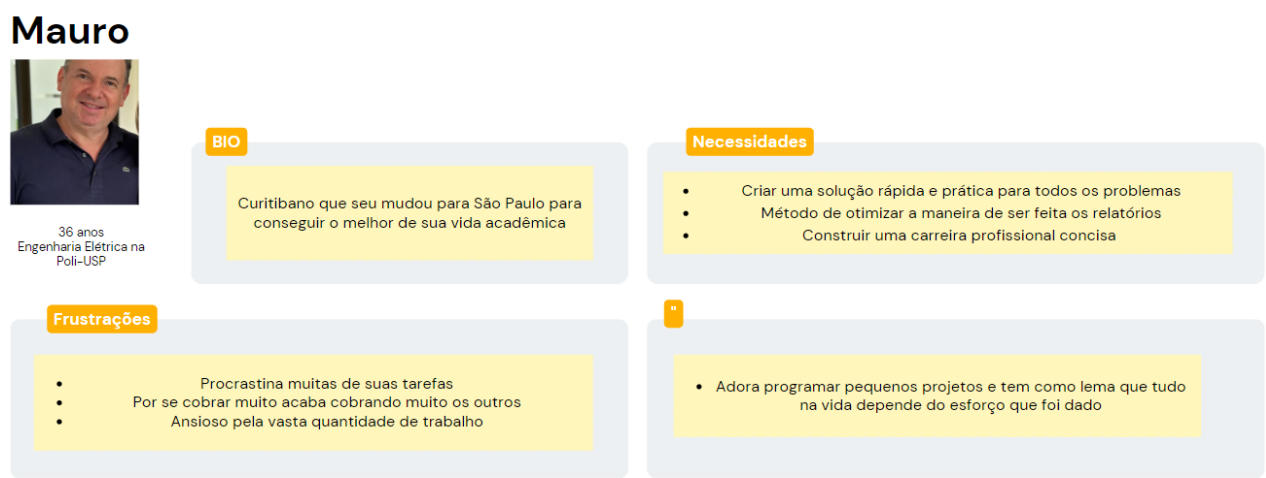
- Muito explosiva e impaciente com seus parceiros de trabalho
- Viciada em trabalhar, que causa muitas horas extras
- Gestão de tempo muito deficiente

"

- Adora otimizar seu trabalho de todas as maneiras, tendo uma relação muito próxima com a tecnologia

Fonte: Elaborado pelos autores.

Figura 7 - Persona 3



Fonte: Elaborado pelos autores.

2.6. Histórias dos usuários (user stories)

Histórias dos usuários são descrições breves, do ponto de vista do usuário, por meio de critérios e testes de aceitação e fornecendo diretrizes para o desenvolvimento e validação de funcionalidades. Realizamos essas tabelas, que estão abaixo deste parágrafo. Essas tabelas (2,3 e 4) foram desenvolvidas em sala de aula com objetivo de pensar em possíveis dificuldades das personas e quais seriam as validações que teriam que ser feitas para reduzir esses impedimentos.

Tabela 2 - História de usuário Mauro

Número	0X1Y		
Grau de Complexidade	5	Grau de prioridade	3
Título	Acesso aos dados com filtragem de prioridade.		
Persona	Mauro, engenheiro elétrico		
História	Eu como pesquisador e engenheiro, quero acessar dados específicos com mais facilidade para agilizar meu trabalho.		

Critérios de Aceitação	CR01 - Conseguir acessar as informações escolhidas através dos filtros. Validação: Aparecer a informação correspondente ao filtro selecionado.	CR02 - As opções de filtro são suficientes para realizar as pesquisas necessárias. Validação: Os filtros aparentes atendem a todas as opções de pesquisa.
Testes de Aceitação	CR01 - <ul style="list-style-type: none"> a. A informação que aparece corresponde ao(s) filtro(s) selecionado(s). <ul style="list-style-type: none"> - Atingiu: Correto. - Não atingiu: Está errado e precisa ser corrigido. 	CR01 - <ul style="list-style-type: none"> a. O usuário não achou o filtro que procurava e teve que procurar os dados de maneira manual. <ul style="list-style-type: none"> Atingiu: Errado, precisa adicionar o filtro necessário. Não atingiu: correto, os filtros disponíveis são suficientes.

Fonte: Elaborado pelos autores

Tabela 3 - História de usuário Cézar

Número	0X2Y	
Grau de Complexidade	1	Grau de prioridade
Título	Pop-ups de exemplos e tutoriais de como navegar no site	
Persona	Cézar, engenheiro mecânico	
História	Eu como um engenheiro mais antigo, gostaria de aprender como eu mexo nas coisas que preciso para o meu trabalho.	
Critérios de Aceitação	CR01 - Depois de logar com e-mail e senha no relatório, o usuário terá acesso a um botão de ajuda em todas as páginas do site. Validação: Os botões atendem os requisitos básicos para poder mexer no site.	
Testes de Aceitação	CR01 - <ul style="list-style-type: none"> a. Não conseguiu acessar a ferramenta básica para poder começar a mexer no site. <ul style="list-style-type: none"> Atingiu: Errado, precisa adicionar. 	

Fonte: Elaborado pelos autores

Tabela 4 - História de usuário Susana

Número	0X3Y		
Grau de Complexidade	3	Grau de prioridade	5
Título	Acesso aos pontos críticos a partir do mapa		
Persona	Susana, Executiva		
História	Eu como executiva e cliente quero acessar os pontos críticos no trajeto para conseguir realizar políticas de prevenção e melhora das ferrovias.		
Critérios de Aceitação	CR01 - Acesso a informações a respeito dos pontos críticos por meio do mapa de trajeto. Validação: Ao clicar nos pontos críticos, informações pertinentes aparecem na tela.		
Testes de Aceitação	CR01 - <ul style="list-style-type: none"> a. Ela clicou num ponto que não é crítico no mapa. Aceitou: Errado, deve ser corrigido Recusou: Correto. b. Ela clicou num dos pontos críticos. Aceitou: Correto, deve mostrar as informações. Recusou: Errado, deve ser corrigido. 		

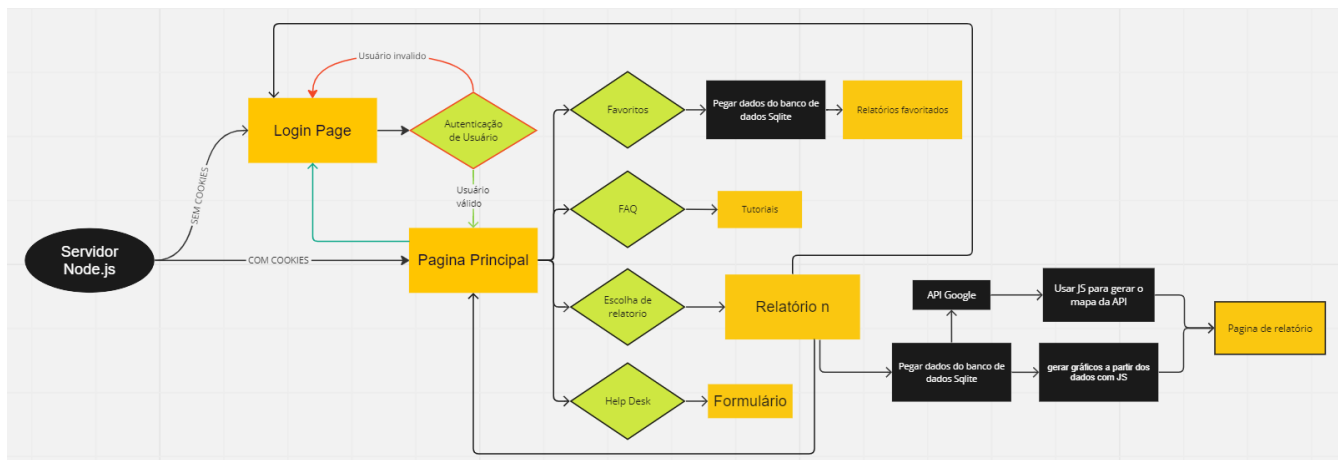
Fonte: Elaborado pelos autores

3. Arquitetura do Sistema

3.1. Módulos do Sistema e Visão Geral (Big Picture)

Os módulos do sistema e visão geral de nossa aplicação WEB foram desenvolvidos em uma atividade em grupo, onde mostramos na figura (Figura 8) os blocos funcionais independentes que compõe o sistema de software e também a interação e fluxos de informação, procura facilitar a compreensão de gestores e desenvolvedores sobre a arquitetura do projeto.

Figura 8 - Diagrama do hardware e software



Fonte(Elaborado pelos autores): https://miro.com/app/board/uXjVMMXUEss=/?share_link_id=477395680222

O Diagrama representando acima (Figura 8) mostra o hardware e software do sistema, mostrando todas as suas funcionalidades e caminhos possíveis ao entrar em nosso servidor, também nesta figura (8) tem todas as tecnologias utilizadas e onde elas estão sendo aplicadas, tendo uma grande importância para o entendimento prévio da página web e como será a sua funcionalidade.

3.2. Tecnologias Utilizadas

Esta tabela (Tabela 3) foi pensada para demonstrar todas as tecnologias utilizadas em nossa aplicação WEB, mostrando o que é, em que é utilizada e a sua versão.

Tabela 3 - Tecnologias Utilizadas

Tecnologia Utilizada	O que é?	Em que é utilizada?	Versão
CSS	CSS Cascading Style Sheets é utilizada para estilizar as páginas criadas através do HTML.	CSS foi usado na parte estética do site, como os ícones, as cores das páginas e os formatos e layouts de páginas.	CSS3
DB Schema	Uma ferramenta que serve para a modelagem de bancos de dados com os diagramas de entidade e relacionamento.	Utilizada para criação de relacionamentos entre as tabelas que vamos criar através do banco de dados.	Versão: 9.3.0
HTML	HTML em inglês Hyper Text Markup Language é usado para marcação	Vamos usar para montar o esqueleto na página WEB para dispor	HTML5

	de textos. Focada em criar conteúdo e estruturar páginas web.	dos ícones e relatórios da melhor maneira possível.	
JavaScript	Linguagem de programação interpretada que é amplamente usada para a web, por ser dinâmico e interativo.	Linguagem de programação que está sendo usada no projeto para conversar entre os bancos e outras ferramentas que estão sendo usadas no projeto.	Não tem.
Node.js	Plataforma de desenvolvimento de criação de aplicativos, API's e a linguagem de programação utilizada é o JavaScript.	Usado como servidor para poder criar as aplicações WEB e a API que precisamos.	Versão: 20.0.0.
DB Browser (SQLite)	Ferramenta de código livre para criar, projetar e editar bancos de dados	Criação e manuseio do banco de dados oferecido pelo parceiro (dados modificados).	Versão 3.12.2
Postman	Facilitador para fazer os testes, depurar e documentar as APIs	Comandos como GET, POST, PUT, DELETE.	Versão 10.13.5
VSCode	Editor de código para entender como NodeJS, HTML, CSS pode ficar	Ensaio e entendimento dos códigos para a prototipação do projeto.	Versão 1.78.0

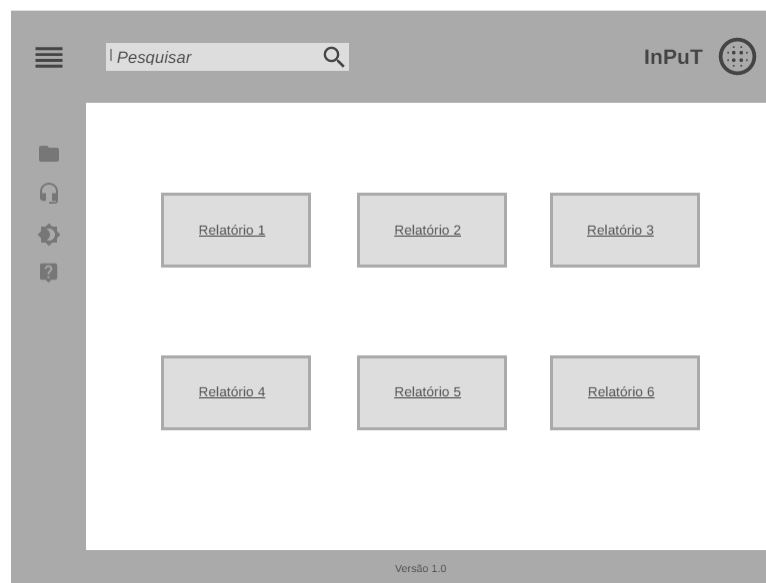
Fonte: Elaborado pelos autores

4. UX e UI Design

4.1. Wireframe

O Wireframe é uma representação visual básica e esquemática de uma página web, cuja utilidade é apresentar a ideia para líderes do projeto e facilitar a comunicação entre designers e desenvolvedores na implementação da página. Como apresentado nas figuras (9 e 7).

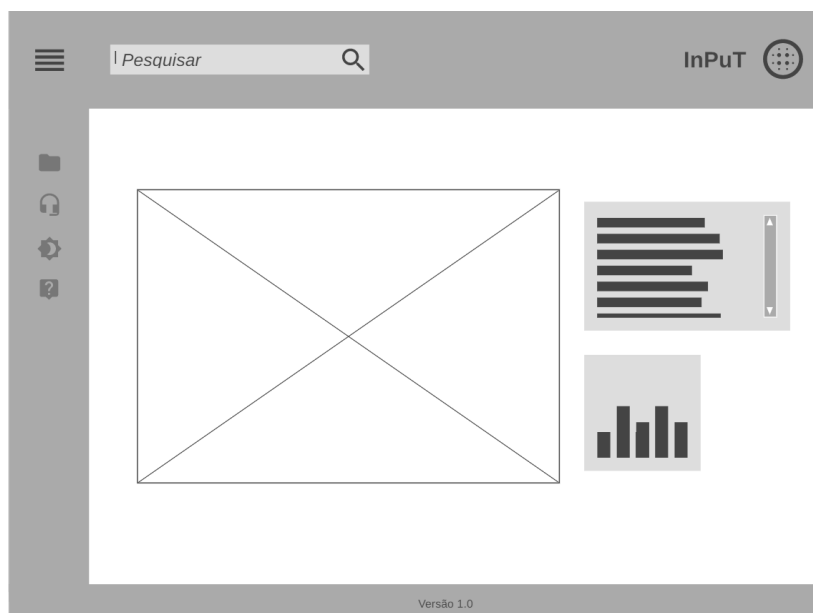
Figura 9 - Tela Inicial



Fonte: Elaborado pelos autores.

A figura 9 é a tela inicial do nosso site, nela é possível acessar a barra de pesquisa na aba superior; na aba lateral encontram-se um ícone que leva ao repositório dos relatórios, um que leva à página de serviço, um que altera o modo da tela entre claro e escuro e um que leva à página de “perguntas frequentes”

Figura 10 - Relatório



Fonte: Elaborado pelos autores.

Na figura 10 é mostrado a aba do repositório, é possível acessar todos os relatórios que estiverem disponíveis na página e, ao clicar em um deles, um mapa da rota ferroviária percorrida pelos trens analisados no relatório é aberta, e nela, pontos críticos do percurso podem ser acessados, revelando informações mais detalhadas sobre aquela região do trilho, e gráficos ou tabelas com informações relevantes.

4.2. Design de Interface - Guia de Estilos

O guia de estilos é um documento desenvolvido pelo grupo que indica e mostra a tipografia, paleta de cores e ícones do projeto como nas figuras 11, 12 e 13, assim representando a identidade visual da página.

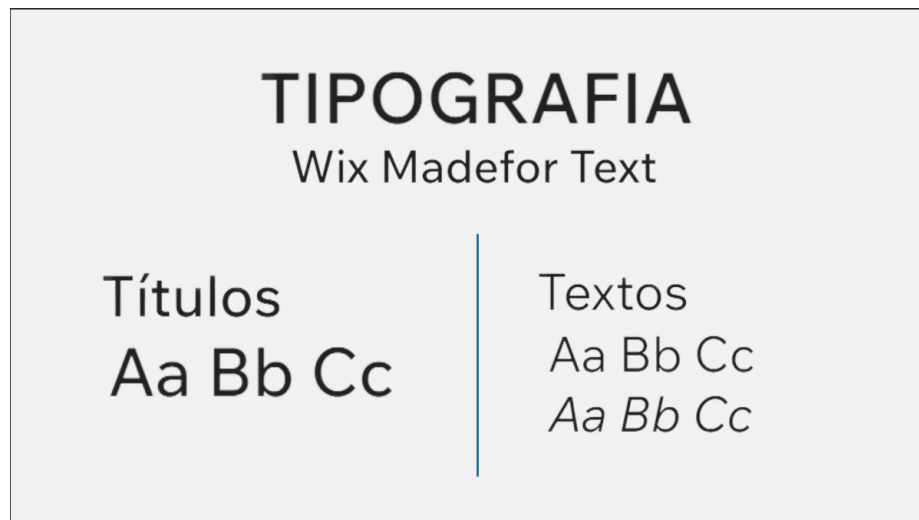
Figura 11 - Paleta de Cores



Fonte: Elaborado pelos autores.

Na figura 11, estão listadas as cores da paleta do projeto nas escalas de cor RGB, CMYK e HEX.

Figura 12 - Tipografia



Fonte: Elaborado pelos autores.

A figura 12 mostra a fonte que está sendo utilizada no site: Wix Madefor Text, disponibilizada pelo site [Google Fonts](#). Essa fonte foi escolhida por seu formato arredondado que traz certa familiaridade e seu visual limpo que torna a leitura mais fácil. Nos títulos, utilizamos a família Medium 500, que é uma versão em negrito desta fonte, e nos demais textos, utilizamos a família Regular 400.

Figura 13 - Ícones



Fonte: Elaborado pelos autores.

Na figura 13, estão os ícones utilizados no site. O uso de ícones facilita a navegação do usuário e diminui sua carga cognitiva, por tornar muito mais rápida a associação de qual botão apertar por assimilação dessas imagens. Para isso, ícones comumente utilizados pela maioria dos sites foram escolhidos: uma casinha representando a página inicial, três barrinhas paralelas para identificar o menu, e assim por diante.

5. Projeto de Banco de Dados

5.1. Modelo Conceitual

O modelo conceitual de um banco de dados é uma representação abstrata dos dados que serão armazenados no banco de dados, descrevendo as entidades e relacionamentos entre elas. Escolhemos utilizar um banco de dados chamado "Rels" que guarda relatórios e seus ID's correspondentes, outro banco de dados chamado "userprefs" que guarda usuários e favoritos (ver Modelo Relacional), e um template para criação de um banco de dado novo por relatório, chamando ReIN, sendo N um número para esse relatório.

As tabelas do banco de dados são:

Aqui está o modelo relacional para o banco de dados "Rels" com uma única tabela "Relatorios", com base nas informações fornecidas:

- Tabela: RELATORIOS
 - ID (chave primária, inteiro, autoincremento)
 - relatorios (inteiro, não nulo)

Neste modelo relacional, a tabela "RELATORIOS" é a única tabela presente e armazena as informações de relatórios, cada um identificado por um ID único. O campo "relatorios" armazena algum valor inteiro associado ao relatório, talvez um indicador ou uma contagem, e é um campo obrigatório.

Aqui está o modelo relacional para o template do banco de dados ReIN utilizado para um relatório, com base nas informações fornecidas:

- Tabela: OCORRENCIA
 - ID_OC (chave primária, inteiro)
 - tipo_oc (texto, tamanho 1, não nulo, verificação para valores 'P' ou 'C')
 - tipo_vagao (texto, tamanho 1, não nulo, verificação para valores 'E' ou 'F')
 - viagem (Integer, não nulo)
 - data_hora (timestamp, não nulo)
 - lat (numérico, não nulo)
 - lon (numérico, não nulo)
 - trecho (texto, não nulo)
 - pos (numérico, não nulo)
 - pv (texto, não nulo)
- Tabela: PICO
 - ID_OC (chave estrangeira referenciando OCORRENCIA.ID_OC)

- ID (chave primária, inteiro)
 - vel (numérico, não nulo)
 - engate (numérico, não nulo)
 - delta_t (numérico, não nulo)
 - act (numérico, não nulo)
 - peg_psi (numérico, não nulo)
- Tabela: CHOQUE
 - - ID_OC (chave estrangeira referenciando OCORRENCIA.ID_OC)
 - ID (chave primária, inteiro)
 - tipo_choque (texto, tamanho 1, não nulo)
 - peg_psi (numérico, não nulo)
 - act (numérico, não nulo)
 - f_max (numérico, não nulo)
 - vel (numérico, não nulo)

Nesse modelo relacional, a tabela "OCORRENCIA" é a tabela principal que armazena as informações gerais sobre cada ocorrência. As tabelas "PICO" e "CHOQUE" estão relacionadas com a tabela "OCORRENCIA" por meio da chave estrangeira "ID_OC", permitindo a associação dessas informações específicas a uma ocorrência. A tabela "PICO" contém detalhes sobre picos relacionados à ocorrência, enquanto a tabela "CHOQUE" armazena informações sobre os choques relacionados à ocorrência. Cada tabela possui sua própria chave primária ("ID" para "PICO" e "CHOQUE") para identificação exclusiva de cada registro na tabela.

- Fora as tabelas relacionadas ao relatório do IPT, ainda há as tabelas que são necessárias para o funcionamento de outras ferramentas do site, tal como a tela de login e os favoritos.

Aqui está o modelo relacional para o banco de dados "userprefs" com as tabelas "favs" e "users", com base nas informações fornecidas:

- Tabela: CHANS
 - chan_id (chave primária, inteiro, autoincremento)
 - user_type (texto, não nulo)
- Tabela: USERS
 - id (chave primária, inteiro, autoincremento)
 - username (texto, não nulo, único)
 - password (texto, não nulo)
- Tabela: FAVS
 - id_user (chave estrangeira, inteiro, referenciando USERS.id)
 - rel_num (inteiro)

- Tabela: COMMENTS
 - comment_id (chave primária, inteiro, autoincremento)
 - chan_id (inteiro, não nulo)
 - rel_id (inteiro, não nulo)
 - content (texto, não nulo)

Neste modelo relacional, a tabela "USERS" é a tabela principal que armazena as informações dos usuários. Cada usuário é identificado por um ID único. Os campos "username" e "password" armazenam o nome de usuário e a senha de cada usuário, respectivamente.

A tabela "FAVS" está relacionada à tabela "USERS" por meio da chave estrangeira "id_user", que referencia o campo "id" da tabela "USERS". Isso permite associar cada favorito a um usuário específico. O campo "rel_num" na tabela "FAVS" provavelmente armazena um identificador para o relatório favorito do usuário.

A tabela "CHANS" está relacionada à tabela "USER" por meio da chave estrangeira "chan_id" que referencia o campo "id" da tabela "USERS". Que permite que um usuário na mesma conta possa ver os favoritos da mesma conta. Nesse sistema, cada usuário pode ter zero ou mais relatórios favoritos, e cada relatório favorito está associado a exatamente um usuário.

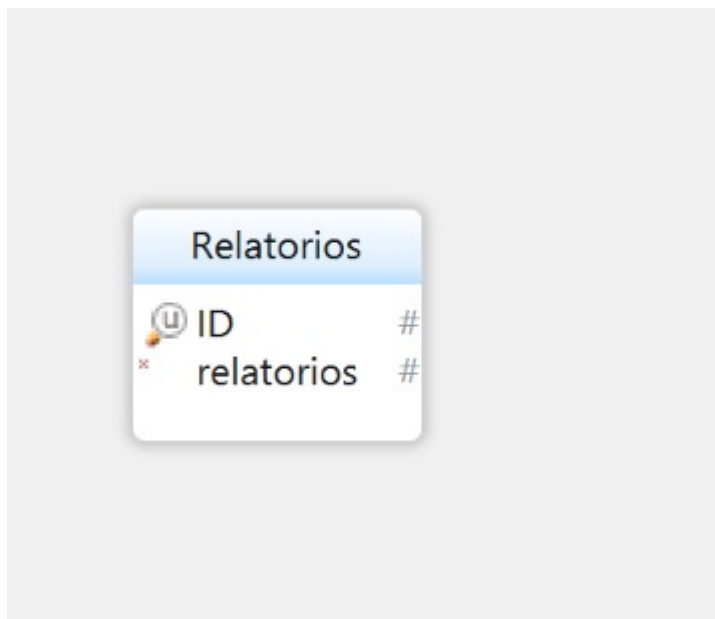
A tabela "COMMENTS" está relacionada à tabela "CHANS" por meio da chave estrangeira "chan_id" que referencia o campo "id" da tabela "USERS". Isso permite que mais de um usuário logado na mesma conta visualize os comentários feitos nela. Nesse sistema, cada usuário pode ter zero ou mais comentários, e cada comentário está associado a exatamente um usuário, autorizando a edição deste individualmente por máquina.

5.2. Modelo Lógico

O modelo lógico (Figuras 14, 15 e 16) é uma representação abstrata e conceitual do sistema cuja função é descrever a estrutura das tabelas, os relacionamentos entre elas, as chaves primárias e estrangeiras. Ou seja, ele traz mais informações que um modelo conceitual, tendo uma análise mais aprofundada do sistema.

- Modelo para "Rels" (Banco de dados que guarda uma lista de relatórios)

Figura 14

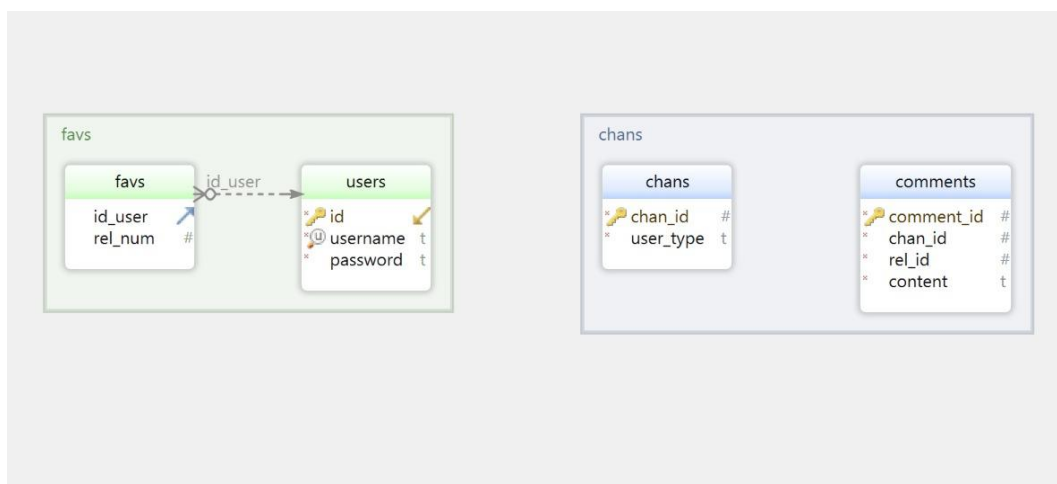


Fonte: Elaborado pelos autores.

A figura 14 mostra a tabela que guarda o número de relatórios que serão inseridos dentro do banco de dados.

- Modelo para “userprefs” (Guarda usuários e favoritos)

Figura 15

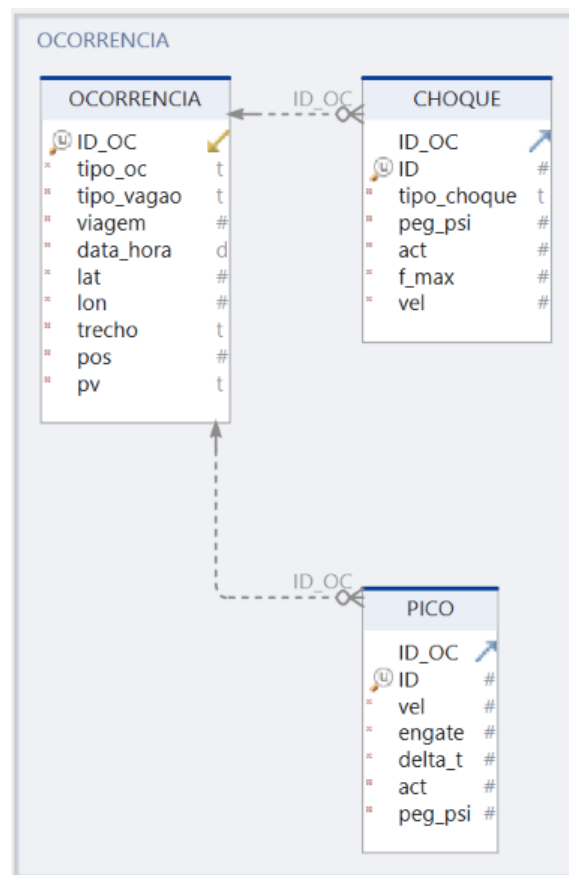


Fonte: Elaborado pelos autores.

A figura 15 mostra o relacionamento entre a tabela de favoritos e a tabela users, indicando que um user pode ter entre 0 e muitos favoritos. Além disso, temos também a tabela chan, que registra o acesso do navegador para a edição de comentários, temos também a tabela comments que registra os comentários em si.

- Modelo para “ReIN” (sendo N o número de um relatório)

Figura 16



Fonte: Elaborado pelos autores.

Na figura 16, mostra-se a tabela Ocorrência, que possui as informações gerais (como data/hora, localização, entre outros). Essa ocorrência pode ser um choque ou um pico, duas outras tabelas que estão ligadas à tabela de ocorrências.

6. Testes de Software

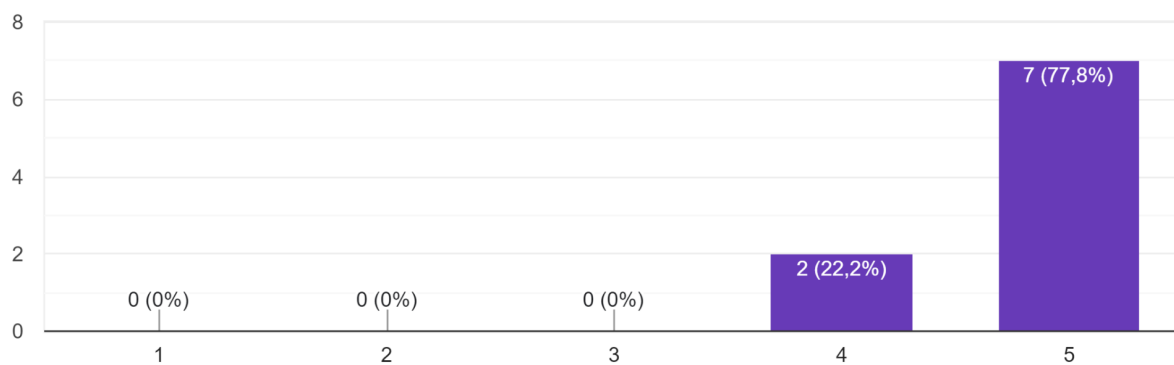
6.1. Teste de Usabilidade

Os testes foram conduzidos por meio de uma atividade interclasses visando verificar e testar a usabilidade, funcionalidade e acessibilidade da aplicação web desempenhando tarefas que os clientes vão fazer. Utilizando como público-alvo alunos do primeiro semestre do Inteli, com faixa etária entre 17 a 20 anos, que tiveram como objetivo avaliar e apontar possíveis melhorias através do *system usability scale* (SUS) apresentado na tabela abaixo.

Figura 17

1 - Eu acho que gostaria de usar esse sistema com frequência.

9 respostas

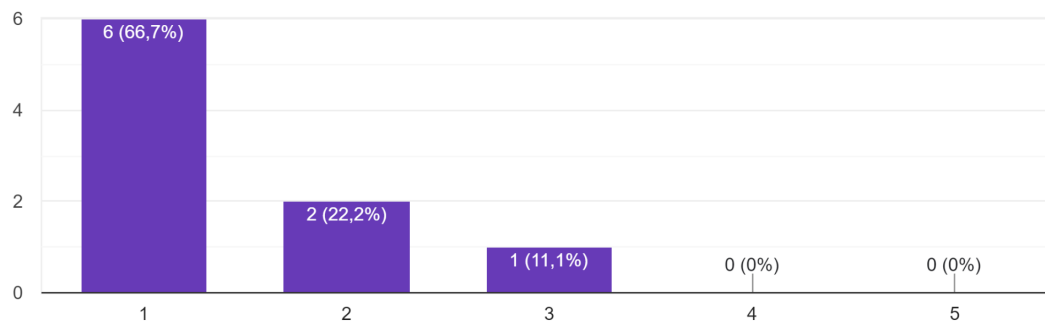


Fonte: Elaborado pelos autores

Figura 18

2 - Eu achei o sistema desnecessariamente complexo.

9 respostas

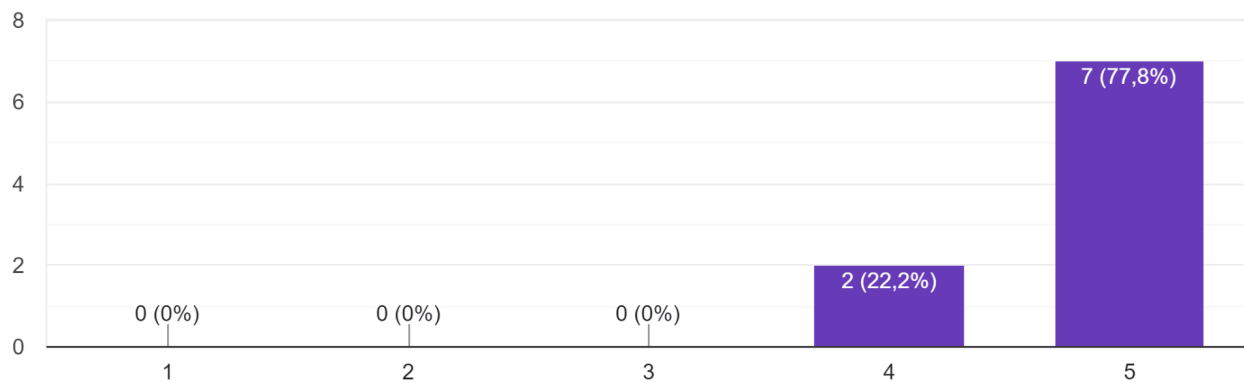


Fonte: Elaborado pelos autores

Figura 19

3 - Eu achei o sistema fácil de usar.

9 respostas

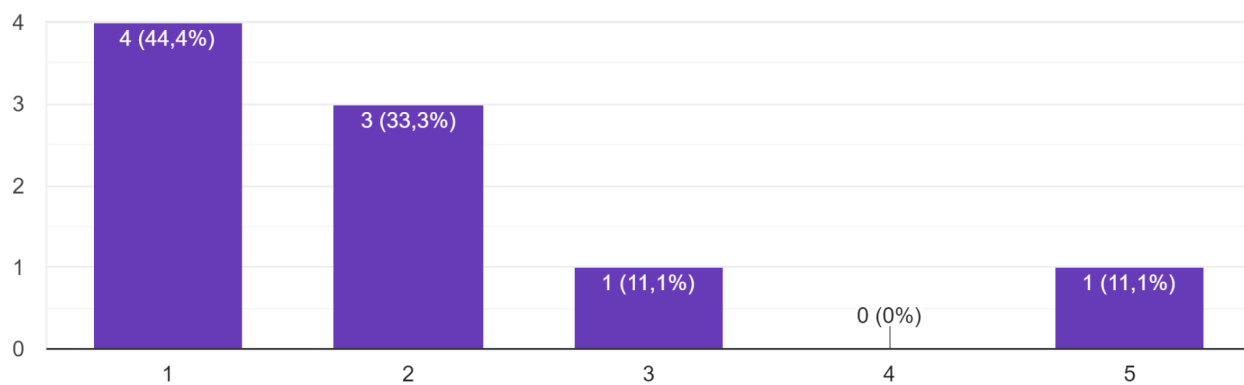


Fonte: Elaborado pelos autores

Figura 20

4 - Eu acredito que precisaria de suporte técnico para usar esse sistema.

9 respostas

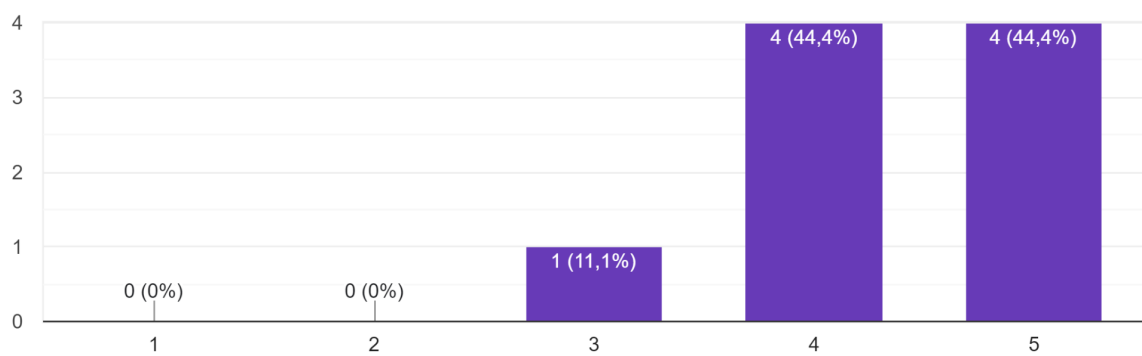


Fonte: Elaborado pelos autores

Figura 21

5 - Eu achei as funções do sistema bem integradas.

9 respostas

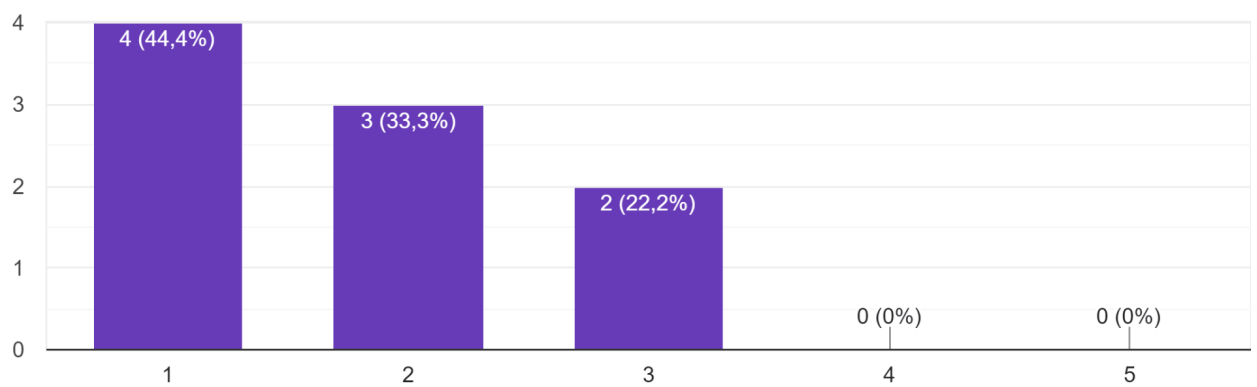


Fonte: Elaborado pelos autores

Figura 22

6 - Eu achei que houve muita inconsistência no sistema.

9 respostas

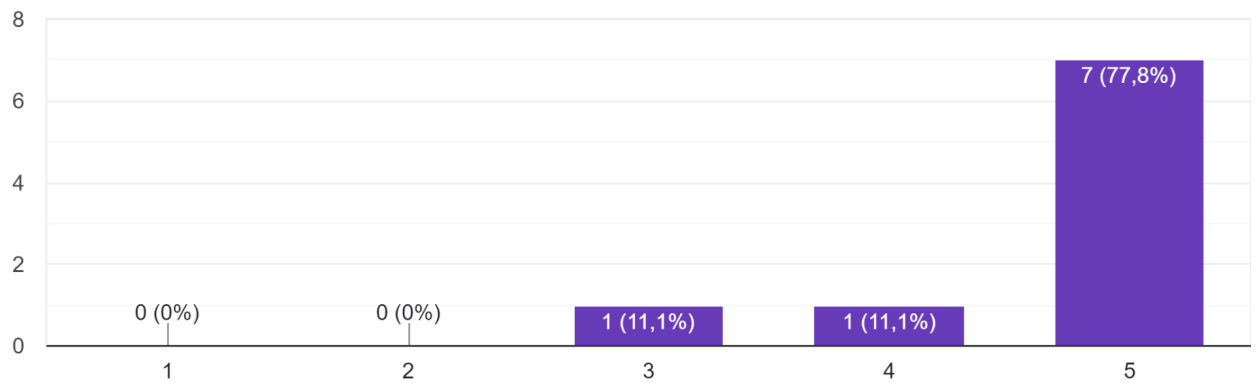


Fonte: Elaborado pelos autores

Figura 23

7 - Eu acredito que a maioria das pessoas aprenderia a usar esse sistema rapidamente.

9 respostas

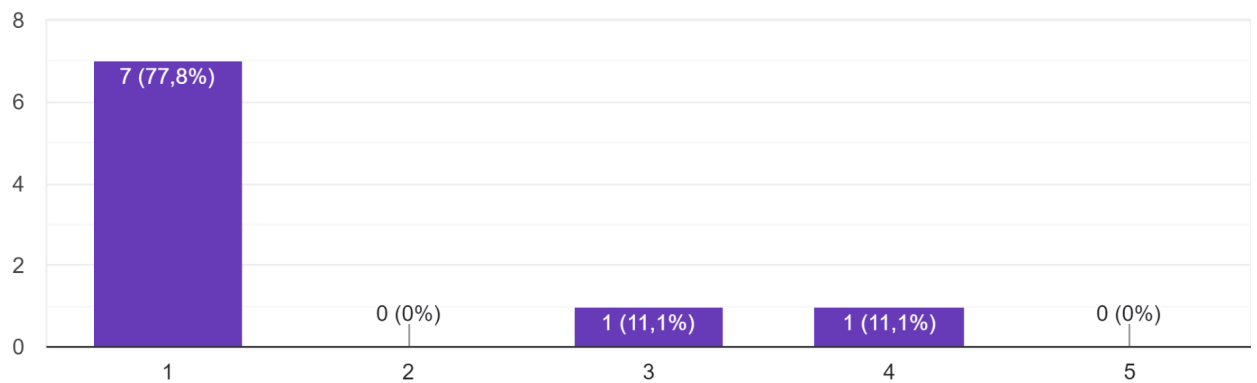


Fonte: Elaborado pelos autores

Figura 24

8 - Eu achei o sistema muito confuso de usar.

9 respostas

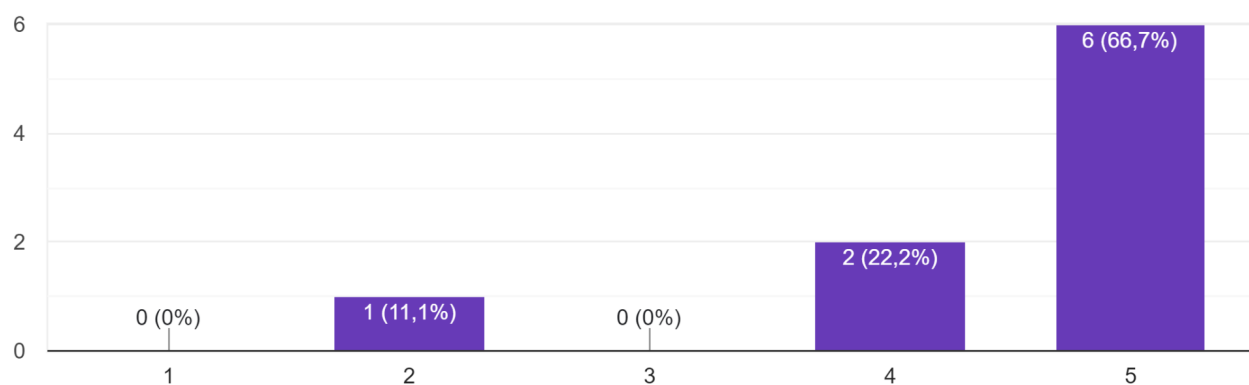


Fonte: Elaborado pelos autores

Figura 25

9 - Eu me senti confiante usando esse sistema.

9 respostas

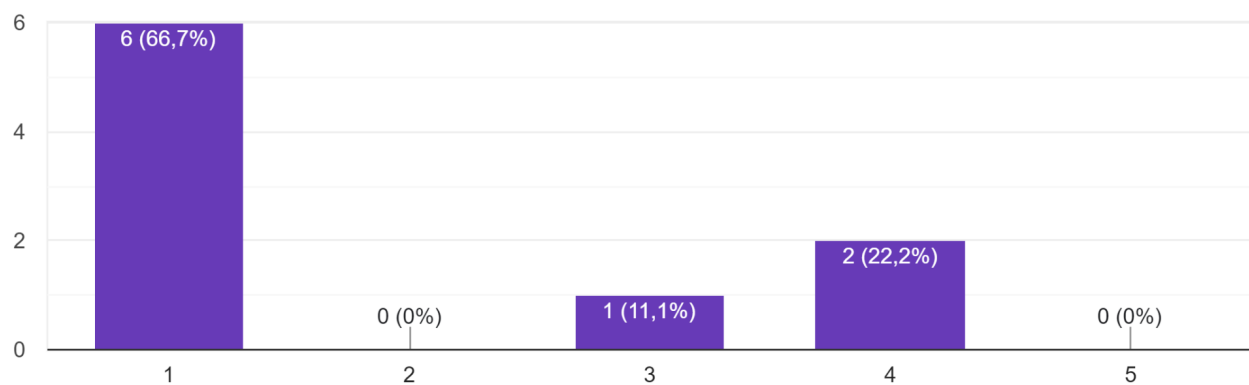


Fonte: Elaborado pelos autores

Figura 26

10 - Eu precisaria aprender muitas coisas novas antes de poder usar esse sistema.

9 respostas



Fonte: Elaborado pelos autores

A partir dos dados coletados através das pesquisas realizadas pelos usuários ao final do teste, foi possível notar que o nosso site foi avaliado com excelência tendo uma nota de 85,9 segundo o método SUS, podendo concluir que a página web está seguindo os quesitos de usabilidade.

Realização:

As sessões foram realizadas nos próprios computadores dos alunos e dentro das salas de aulas do Inteli. Os *testers* estavam acompanhados por alunos do grupo, que não podia interferir em nenhum momento, onde sua função era explicar os requerimentos da empresa e o que era esperado do usuário conseguir com a versão 4.0 do site. O aluno do grupo pôde analisar e anotar todas as dificuldades, facilidades e erros encontrados pelo usuário na página web e com isso preencher um formulário ao final onde anotava todas as dificuldades e as metas preestabelecidas feitas com sucesso ou fracasso. Seguem as tabelas(1, 2 e 3) com essas informações:

Tabela 1 - Perfil dos participantes e os objetivos esperados

PARTICIPANTES					QUEST FINAL			
	Nome	Perfil	Curso	Turma	Acessou o relatório?	Conseguiu baixar o relatório?	Favoritou os relatórios?	Abriu os mapas?
1	Kaylane Vasconcelos	Estudante	Sistemas de Informação	6	Conseguiu acessar	sim	sim, mas com dificuldade	sim
2	Henrique Ottoboni	Estudante	Engenharia de Software	6	Conseguiu acessar	Sim, porém clicou primeiro dentro do relatórios (que não funciona e só depois na parte de relatórios)	Sim, porém só paracia quando voltasse para o painel de controle	sim
3	Kaiane	Estudante	Engenharia de Software	6	Conseguiu acessar	Sim, porém clicou primeiro dentro do relatórios (que não funciona e só depois na parte de relatórios)	Sim, porém só paracia quando voltasse para o painel de controle	sim
4	Michel	Estudante	Sistemas de Informação	9	Conseguiu acessar	Sim	Sim	Sim
5	Isabelle Oliveira	Estudante	Sistemas de Informacao	7	Conseguiu acessar	Sim	sim, com dificuldade	sim
6	Pablo Ruan	Estudante	Engenharia da Computacao	8	Conseguiu acessar	sim	sim	sim
7	Ólin Costa	Estudante	Engenharia da Computacao	8	Conseguiu acessar	sim	sim, mas precisava atualizar a página	sim
8	luiza	Estudante	Sistemas de Informação	8	Conseguiu	Sim, porém clicou primeiro dentro do relatórios (que não funciona e só depois na parte de relatórios)	Sim	Sim
9	Bruno	Estudante	Engenharia da Computacao	8	Conseguiu	Sim, porem questionou o nome pouco intuitivo(talvez de Rel para arrhual)	Sim(Sentiu falta de feedbacks)	Sim, mas não gostou qu tudo recarregava

Fonte: Elaborado pelos autores

Tabela 2 - Resultado da tarefa de baixar um relatório

RESULTADOS POR TAREFA						
1.Baixar um relatório			RESULTADO GERAL	RESULTADO POR ETAPA		
#	Nome	Perfil	RESULTADO DA TAREFA	ACESSAR A PÁGINA DE RELATÓRIOS	LOCALIZAR O BOTÃO	BAIXAR
1	Kaylane Vasconcelos	Estudante	sucesso	sucesso	sucesso	sucesso
2	Henrique Ottoboni	Estudante	Cumpriu com um pouco de dificuldade (usou o botão dentro do relatório primeiro)	sucesso	sucesso	sucesso
3	Kaiane	Estudante	Cumpriu com um pouco de dificuldade (usou o botão dentro do relatório primeiro)	sucesso	sucesso	sucesso
4	Michel	Estudante	sucesso	sucesso	sucesso	sucesso
5	Isabelle Oliveira	Estudante	sucesso	sucesso	sucesso	sucesso
6	Pablo Ruan	Estudante	sucesso	sucesso	sucesso	sucesso
7	Ólin Costa	Estudante	sucesso	sucesso	sucesso	sucesso
8	luiza	Estudante	sucesso	sucesso	sucesso	sucesso
9	Bruno	Estudante	sucesso	sucesso	sucesso	sucesso

Fonte: Elaborado pelos autores

Tabela 3 - Resultados da tarefa de favoritar um relatório

2. Favoritar um relatório			RESULTADO GERAL	RESULTADO POR ETAPA			
#	Nome	Perfil	RESULTADO DA TAREFA	ACESSAR A PÁGINA	LOCALIZAR O BOTÃO DE FAVORITOS	FAVORITAR	LOCALIZAR O FAVORITO (página inicial e menu)
1	Kaylane Vasconcelos	Estudante	Conseguiu com dificuldade	sucesso	sucesso	Conseguiu com dificuldade	Conseguiu com dificuldade
2	Henrique Ottoboni	Estudante	sucesso	sucesso	sucesso	sucesso	conseguiu com dificuldade
3	Kaiane	Estudante	Cumprir com um pouco de dificuldade	sucesso	sucesso	sucesso	Cumprir com um pouco de dificuldade
4	Michel	Estudante	sucesso	sucesso	sucesso	sucesso	conseguiu com dificuldade
5	Isabelle Oliveira	Estudante	Cumprir com dificuldade	sucesso	sucesso	conseguiu com dificuldade	fracasso
6	Pablo Ruan	Estudante	Cumprir com dificuldade	sucesso	sucesso	conseguiu com dificuldade	fracasso
7	Ólin Costa	Estudante	sucesso	sucesso	sucesso	conseguiu com dificuldade	conseguiu com dificuldade
8	Iulza	Estudante	Cumprir com dificuldade	sucesso	sucesso	conseguiu com dificuldade	sucesso
9	Bruno	Estudante	Cumprir com dificuldade	sucesso	sucesso	conseguiu com dificuldade	conseguiu com dificuldade

Fonte: Elaborado pelos autores

Segue o link para a pré-visualização da tabela no Excel: [Cópia de Tabulação de teste de usabilidade](#)

Com base na nossa análise feita a partir das informações adquiridas nas tabelas acima, conseguimos adquirir um nível extremamente satisfatório no nosso site, podendo ver que nossos usuários atingiram com sucesso a maioria das tarefas dadas como objetivos anteriormente para os *testers*. Todavia recebemos feedbacks sendo a maioria deles dizendo que nosso site não é comunicativo, ou seja, quando ocorre uma ação em um botão não existe um feedback visual que indica o sucesso dessa atividade, como grupo já foi pensado em uma resolução para esse problema, utilizando de alertar e pop-ups na tela do usuário, indicando o sucesso ou fracasso da ação.

Referências

Bootstrap · The most popular HTML, CSS, and JS library in the world, 2011. Disponível em: <https://getbootstrap.com>

IPT: Instituto de Pesquisa e Tecnologia. Disponível em: <https://www.ipt.br>

Feather. Feather - Simply beautiful open source icons. Disponível em: <https://feathericons.com/>

Apêndice

Relatório de Criação de Endpoints - 13/05/2023

Introdução

Este relatório tem como objetivo apresentar e definir a criação de endpoints no desenvolvimento de uma aplicação web.

Desenvolvimento

O primeiro endpoint HTTP GET é para a página inicial, que é acessada quando um usuário faz uma solicitação GET para a raiz do servidor. Quando isso acontece, o servidor envia um arquivo HTML como resposta.

O segundo endpoint HTTP GET é para a página de informações sobre o pico, que é acessada quando um usuário faz uma solicitação GET para a rota `/graphPico/:id/:ocur`. Neste endpoint, os parâmetros da rota `:id` e `:ocur` são usados para determinar qual tabela e coluna do banco de dados são consultados.

Se o valor de `:id` for 1, a variável `'table'` é definida como `"OCORRENCIAS_PICO1"`. Em seguida, dependendo do valor de `:ocur`, uma consulta SQL diferente é feita no banco de dados para obter os dados da coluna correspondente da tabela `"OCORRENCIAS_PICO1"`. A resposta da consulta é enviada como uma resposta JSON.

Se o valor de `:id` for 2, a variável `'table'` é definida como `"OCORRENCIAS_PICO2"`. Novamente, dependendo do valor de `:ocur`, uma consulta SQL diferente é feita no banco de dados para obter os dados da coluna correspondente da tabela `"OCORRENCIAS_PICO2"`. A resposta da consulta é enviada como uma resposta JSON.

O terceiro endpoint é definido para a rota raiz (`/`) e usa o método `sendFile` do Express.js para enviar o arquivo HTML da página inicial como resposta. Esse endpoint é acessado quando o usuário acessa a página principal do aplicativo.

O quarto endpoint é definido para a rota `/graphPico/:id/:ocur` e usa os parâmetros de URL `:id` e `:ocur` para determinar qual tabela do banco de dados deve ser consultada e quais colunas devem ser retornadas na resposta JSON. O endpoint usa condicionais `if` para determinar qual consulta SQL deve ser executada com base nos valores dos parâmetros de URL.

O quinto endpoint é definido para a rota `/graphsE/:ocur/:id` e também usa os parâmetros de URL `:id` e `:ocur` para determinar qual tabela do banco de dados deve ser consultada e quais colunas devem ser retornadas na resposta JSON. O endpoint usa condicionais `if` semelhantes aos usados no

segundo endpoint para determinar qual consulta SQL deve ser executada com base nos valores dos parâmetros de URL.

O sexto endpoint executa a função `app.get('/graphsF/:ocur/:id', ...)`, que define uma rota que manipula as requisições GET para `/graphsF/:ocur/:id`. O `'occur'` e `'id'` são parâmetros de URL que são extraídos da rota usando `req.params`.

Dependendo do valor de `ocur`, a função executa uma consulta SQL diferente no banco de dados e retorna os resultados em formato JSON usando `res.json()`. As consultas SQL estão buscando diferentes colunas de diferentes tabelas que correspondem à ocorrência (`id`) e ao tipo de dado (`ocur`) que está sendo requisitado.

O sétimo endpoint recebe um parâmetro `impact` e executa uma consulta SQL para buscar todos os dados da tabela E, dependendo do valor do parâmetro. Os resultados são retornados em JSON.

O oitavo endpoint é semelhante à rota anterior, mas busca dados de outra tabela, a F.

O nono endpoint recebe dados do corpo da solicitação HTTP e insere esses dados em uma tabela no banco de dados, que coleciona os dados favoritados pelo usuário. Se ocorrer algum erro durante a inserção, uma mensagem de erro será exibida no console.

Conclusão

Com base nos endpoints apresentados, é possível concluir que o sistema permite a obtenção de informações sobre ocorrências de choque de duas viagens diferentes, identificadas como E e F.

Os endpoints `/graphsE` e `/graphsF` retornam dados específicos para as ocorrências de cada viagem, tais como a máxima força de impacto, atividade gerada e a energia potencial gravitacional da posição inicial à posição final. Já os endpoints `/tableE` e `/tableF` fornecem a lista completa de todas as ocorrências de cada viagem, com detalhes como o número de identificação, data, hora e local da ocorrência.

Além disso, o sistema também permite que os usuários salvem relatórios específicos como favoritos, por meio do endpoint `/addFav`. Com isso, é possível acessar rapidamente as informações mais importantes para futuras referências.

Em resumo, os endpoints do sistema permitem a obtenção de informações detalhadas sobre as ocorrências de choque de viagens distintas, além de permitir a marcação de relatórios favoritos para fácil acesso posterior.

Introdução:

Este relatório tem como objetivo apresentar e definir a criação de endpoints no desenvolvimento de uma aplicação web utilizando a biblioteca Express.js. Os endpoints são responsáveis por definir as rotas e as ações correspondentes que o servidor realizará quando receber requisições HTTP específicas. Serão abordadas as partes de download, upload, login, relatórios, favoritos e o CRUD de comentários na parte do backend, enquanto no frontend os endpoints estão definidos em um arquivo de roteamento chamado `router.js` e serão descritos a seguir, juntamente com sua lógica de funcionamento e regras de acesso.

Desenvolvimento:

Backend:

1. Download:

O endpoint de download foi implementado utilizando o módulo `./down_up/download`.

Este código descreve um endpoint em Node.js para baixar um arquivo zip. Para isso, ele importa o módulo `"path"`, que é uma biblioteca interna do Node.js para manipulação de caminhos de arquivos e diretórios. Constrói o caminho do arquivo zip usando o método `"path.join"` — o caminho é construído concatenando as partes: `"/"` (indicando o diretório atual), `"Backend"` (nome do diretório), `"archives"` (nome do subdiretório), `"Rel"+Num` (nome do diretório específico do relatório com base no número) e, por fim, `"Rel.zip"` (nome do arquivo zip).

O ponto crucial do código é o método `"res.download"`, usado para iniciar o download do arquivo. Ele recebe o caminho do arquivo zip como primeiro parâmetro e uma função de callback como segundo parâmetro. A função de callback é executada após o término do download ou se ocorrer algum erro durante o processo.

2. Upload:

O endpoint de upload foi criado com o objetivo de permitir o envio de arquivos zip para o servidor. Ele utiliza o módulo `./down_up/handleUpload` para lidar com o processo de upload. O arquivo enviado é salvo na pasta `"uploads/"` no servidor.

Para ser corretamente configurado, o arquivo deve seguir a seguinte estrutura:

Rel.zip				
	Rel			
		Viagens		
			E ou F	
				Choque_1
				Choque_2
				Pico

* É recomendado também que o pdf do relatório esteja dentro da pasta `"Rel"`.

3. Login:

Foram implementados vários endpoints relacionados ao login. O endpoint `"/login"` recebe uma requisição do tipo POST para autenticar o usuário. O endpoint `"/logout"` é acessado via GET para fazer o logout do usuário. O endpoint `"/get_user"` é utilizado para obter informações sobre o usuário atualmente autenticado. O endpoint `"/get_chan_id"` retorna o ID do canal. O endpoint `"/chan_manager"` lida com a criação ou renovação de um cookie chamado `"chan_id"` com base na existência desse cookie na requisição.

4. Relatórios:

Vários endpoints foram criados para lidar com diferentes tipos de relatórios e gráficos. Por exemplo, o endpoint `"/graphsPico/:viagem/:vagao/:ocur/:RelNum"` é acessado via GET e recebe parâmetros para fornecer informações específicas sobre um determinado relatório e gráfico. Outros endpoints como `"/graphsE/:viagem/:ocur/:id/:RelNum"` e `"/graphsF/:viagem/:ocur/:id/:RelNum"` também retornam dados específicos das tabelas E e F, respectivamente. Além disso, existem endpoints para visualizar mapas (`"/mapE/:viagem/:id/:RelNum"` e `"/mapF/:viagem/:id/:RelNum"`) e um endpoint para obter informações sobre a quantidade total de relatórios (`"/quantity"`).

5. Favoritos:

O endpoint `"/addFav/:rel"` permite adicionar um relatório aos favoritos através de uma requisição POST. O endpoint `"/seeFav"` retorna os favoritos atuais via GET. O endpoint `"/deleteFav/:id"` é utilizado para deletar um registro de favorito específico através de uma requisição DELETE. Além disso, existem endpoints para baixar arquivos relacionados aos favoritos (`"/download/:id"` e `"/downloadpdf/:id"`).

6. Comentários:

Foram implementados quatro endpoints relacionados estruturados como CRUD (Create, Read, Update, Delete) de comentários. O endpoint `"/comment_update"` recebe uma requisição POST para atualizar um comentário. O endpoint `"/comment_get"` é utilizado para obter comentários específicos. O endpoint `"/comment_delete"` permite deletar um comentário através de uma requisição DELETE. O endpoint `"/comment_insert"` recebe uma requisição POST para inserir um novo comentário.

Frontend:

1. Página inicial do aplicativo:

Este endpoint é responsável por lidar com a rota raiz do aplicativo. Primeiro, o middleware `'loggedIn'` é executado para verificar se o usuário está autenticado. Em seguida, é feita uma verificação se o cookie `'remember-login'` existe. Se o cookie existir, o usuário é redirecionado para a página `'index.html'` localizada na pasta `'./Frontend/public'`. Caso contrário, o usuário é redirecionado para a página de login.

2. Página de login:

Descrição: Esse endpoint lida com a rota `'/login'`. Quando um cliente faz uma solicitação GET para essa rota, o servidor envia o arquivo `'login.html'` localizado na pasta `'./Frontend/public'` como resposta.

3. Página de relatórios

Esse endpoint lida com a rota '/reports/:id', onde ':id' é um parâmetro dinâmico representando o ID do relatório. O middleware 'loggedIn' é executado para verificar se o usuário está autenticado. Se o cookie 'remember-login' existir, o servidor envia o arquivo 'Rel.html' localizado na pasta './Frontend/public/reports' como resposta. Caso contrário, o usuário é redirecionado para a página de login.

4. Página de lista de relatórios

Esse endpoint lida com a rota '/relatorios'. Quando um cliente faz uma solicitação GET para essa rota, é verificado se o cookie 'remember-login' existe. Se o cookie existir, o servidor envia o arquivo 'relatorios.html' localizado na pasta './Frontend/public' como resposta. Caso contrário, o usuário é redirecionado para a página de login.

5. Página de upload

Esse endpoint lida com a rota '/upload'. O middleware 'loggedIn' é executado para verificar se o usuário está autenticado. Se o cookie 'remember-login' existir, o servidor envia o arquivo 'upload.html' localizado na pasta './Frontend/public' como resposta. Caso contrário, o usuário é redirecionado para a página de login.

Conclusão:

Neste relatório, foram descritos os endpoints criados no aplicativo web utilizando o framework Express. Cada endpoint possui uma função específica e regras de acesso. Esses endpoints lidam com o download e upload de arquivos, comentários gerados dentro do site, a rota inicial, página de login, páginas de relatórios, lista de relatórios e página de upload. Eles garantem que o usuário esteja autenticado antes de permitir o acesso a determinadas páginas, redirecionando-o para a página de login quando necessário. Esses endpoints são essenciais para o funcionamento adequado do aplicativo e proporcionam uma experiência segura e fluida para os usuários.

Segue abaixo o link para a visualização dos Endpoints via Postman:

[Endpoints](#)