TECHS – Challenge 2023

java

Ser acessível é abrir portas para um mundo onde todos têm igualdade de oportunidades, independentemente das suas diferenças.

Equipe e Responsabilidades:

Cauã Ferrigoli Loureiro – RM 553093

Claudio Silva Bispo – RM 553472

Sara Ingrid da Silva Pereira – RM 554021

Sumário

[Introdução 4](#_Toc148371624)

[Definição do projeto 4](#_Toc148371625)

[Escopo do projeto 5](#_Toc148371626)

[README 6](#_Toc148371627)

[Design do Diagrama de classe 7](#_Toc148371628)

[Entidades 8](#_Toc148371629)

[Atributos e métodos 8](#_Toc148371630)

[1. Usuário: 8](#_Toc148371631)

[2. Gerenciamento de Banco de Dados: 8](#_Toc148371632)

[3. Monitoramento da Jornada do Usuário: 9](#_Toc148371633)

[4. Atualização do Conteúdo ou Sinalização de Melhorias: 9](#_Toc148371634)

[5.Idioma de Interação: 10](#_Toc148371635)

[6.Segurança 10](#_Toc148371636)

[7. Acesso 10](#_Toc148371637)

[Funcionário 11](#_Toc148371638)

[8. Plataformas 11](#_Toc148371639)

[Relacionamentos 12](#_Toc148371640)

[Usuário x Plataforma de login: 12](#_Toc148371641)

[Monitoramento da Jornada do Usuário: 12](#_Toc148371642)

[Atualização do Conteúdo ou Sinalização de Melhorias: 12](#_Toc148371643)

[Idioma de Interação: 12](#_Toc148371644)

[Segurança: 12](#_Toc148371645)

[Login\_Acesso: 13](#_Toc148371646)

[Banco de dados 13](#_Toc148371647)

[Segurança x Login 13](#_Toc148371648)

# Introdução

Imagine um mundo digital onde todos têm acesso fácil e igualitário à informação, independentemente de suas necessidades. Este é o visionário projeto que estamos empenhados em tornar realidade. Como acessar um site e poder escolher o tema? Como ler as informações contidas sem ter que dar um zoom na tela pois a fonte está pequena ou com uma cor que dificulta leitura e atrapalha o entendimento. Já pensou em fazer uma busca usando áudio ao invés de texto?

Nosso projeto tem o objetivo de criar facilidade, simplicidade e incluir suportes que vão ajudar a qualquer tipo de pessoa a localizar um conteúdo com facilidade e sem burocracia.

# Definição do projeto

Nosso projeto visa criar um site altamente acessível, que oferece uma experiência inclusiva para todos os usuários, sendo eles com ou sem deficiência, eles tendo ou não conhecimento sobre os produtos da empresa foco, independentemente de suas necessidades e capacidades. Ele integra funcionalidades avançadas, como autenticação simplificada através de plataformas populares (G-mail, LinkedIn, GitHub, Outlook, Apple), formulários de contato adaptáveis para entrada de texto ou voz, e um chatbot inteligente para assistência instantânea. Busca por texto e voz, leitura do conteúdo da tela, definição do idioma em várias linguagens, escolha do tema da preferência da pessoa.

Queremos atuar nas funcionalidades:

**Autenticação Simplificada:**

Os usuários podem acessar o site facilmente através de suas contas em plataformas existentes, como G-mail, LinkedIn, GitHub, Outlook e Apple.

**Formulários de Contato Adaptáveis:**

O site oferece formulários de contato intuitivos que podem ser preenchidos por texto ou voz, proporcionando uma experiência inclusiva e eficiente.

**Chatbot Inteligente:**

Um chatbot integrado permite aos usuários obter respostas às suas perguntas através de texto ou voz, proporcionando suporte instantâneo e sem a necessidade de interação humana.

**Leitura de Tela e Legendas:**

O site é completamente compatível com leitores de tela, garantindo que usuários com deficiência visual possam navegar com facilidade. Além disso, os vídeos contêm legendas incorporadas ou disponíveis separadamente para garantir a acessibilidade total.

**Monitoramento da Jornada do Usuário:**

O site rastreia a jornada do usuário, fornecendo insights sobre o que foi pesquisado, quanto tempo foi gasto navegando e quais conteúdos foram acessados.

**Seleção de Idioma Personalizada:**

Os usuários têm a opção de escolher o idioma de apresentação dos conteúdos, permitindo que leiam ou ouçam em seu idioma nativo ou em um idioma que estão aprendendo.

**Acesso para Funcionários:**

Os funcionários têm acesso a um painel intuitivo que permite a consulta ao banco de dados. Eles podem entender o que os clientes procuraram e quanto tempo passaram navegando, além de verificar a atualização de conteúdos e responder rapidamente às necessidades dos usuários.

Benefícios:

Estamos buscando não apenas atender aos padrões de acessibilidade, mas também promover uma experiência inclusiva e agradável para todos os usuários. Proporcionando igualdade de acesso à informação e interação na web simples, dinâmica e rápida.

# Escopo do projeto

A linguagem Java será a linguagem principal para o desenvolvimento do backend do site e implementação de funcionalidades do lado do servidor. Vamos conseguir implementas com essa linguem a parte de autenticação do login, gerenciamento do banco de dados, desenvolvimento dos formulários para que possam ser preenchidos por texto ou voz, leitura da tela, e parte de segurança.

# README

# Design do Diagrama de classe



# Entidades

1. Usuário: Autenticação do usuário
2. Banco de dados: Gerenciamento do banco de dados para consultas e envio de informações para o frontEnd
3. Jornada do usuário: Monitorar a jornada do usuário durante a navegação do site
4. Atualização do conteúdo ou sinalização de melhorias
5. Idioma de interação: Definição do idioma de interação no site e nos bancos de dados
6. Segurança: Desde autenticação do acesso, até o controle de visualização de conteúdos.
7. Login de acesso
8. Funcionário
9. Plataformas

# Atributos e métodos

## 1. Usuário:

**Entidade**: Usuário

**Atributos**:

* ID (Inteiro)
* Nome (String)
* email (String)
* Senha (String)
* Plataforma (String)

**Métodos**:

* autenticar(email, senha): Autentica um usuário com base no e-mail e senha fornecidos. Uso das aplicações como G-mail, Linkedin, entre outros.
* registrar(novoUsuario): Registra um novo usuário no sistema.
* dataHora(guardar informações de quando ocorreu)

## 2. Gerenciamento de Banco de Dados:

**Entidade**: Banco de Dados

**Atributos**:

* Conexão (fazer a conexão com o banco de dados – Aqui podemos usar o python para auxiliar no processo de API por exemplo)

**Métodos**:

* conectar(): Estabelece uma conexão com o banco de dados.
* desconectar(): Fecha a conexão com o banco de dados.
* executarConsulta(query): Executa uma consulta SQL no banco de dados.
* executarAtualizacao(query): Executa uma atualização no banco de dados (inserção, atualização, exclusão). Aqui depende do nível de acesso.

## 3. Monitoramento da Jornada do Usuário:

**Entidade**: Jornada do Usuário

**Atributos**:

* ID (Inteiro)
* ID do Usuário (Inteiro)
* Páginas Visitadas (Strings)
* Tempo Gasto (String)

**Métodos**:

* rastrearInteracao(usuarioID, paginaVisitada, tempoGasto): Registra a interação do usuário com o site.

## 4. Atualização do Conteúdo ou Sinalização de Melhorias:

**Entidade**: Atualizacao\_conteudo

**Atributos**:

* ID (Inteiro)
* Tipo de Atualização (String, por exemplo: "Novo Conteúdo", "Atualização de Informações", "Melhoria de Acessibilidade")
* Descrição (String)
* Data da Atualização (Data e Hora)

**Métodos**:

* registrarAtualizacao(tipo, descricao): Registra uma nova atualização de conteúdo no sistema.
* listarAtualizacoes(): Lista todas as atualizações registradas.

## 5.Idioma de Interação:

**Entidade**: IdiomaInteracao

**Atributos**:

* ID do Usuário (Inteiro)
* Idioma Selecionado (String, código do idioma, por exemplo: "pt-br", "en-us")

**Métodos**:

* selecionarIdioma(usuarioID, idioma): Permite ao usuário selecionar o idioma de interação preferencial.
* obterIdioma(usuarioID): Retorna o idioma selecionado para um usuário específico.

## 6.Segurança

Para parte de segurança da aplicação:

**Entidade**: Seguranca

**Atributos**:

* ID (Inteiro)
* ID do Usuário (Inteiro)
* Nível de Acesso (String)
* Token de Autenticação (String)
* Último Acesso (String)

**Métodos**:

* gerarToken(): Gera um novo token de autenticação para o usuário.
* verificarAcesso(token): Verifica se o token fornecido é válido e corresponde a um usuário autenticado.

## 7. Acesso

**Entidade**: Acesso

**Atributos**:

* ID (Inteiro)
* ID do Usuário (Inteiro)
* Data do Acesso (String))
* IP de Acesso (String, possível endereço IP do cliente?)

**Métodos:**

* registrarAcesso(usuarioID, enderecoIP): Registra o acesso de um usuário ao sistema, incluindo data e IP.

## Funcionário

**Entidade:** Funcionário

**Atributos:**

* ID (Inteiro)
* Nome (String)
* E-mail (String)
* Senha (String)

**Métodos:**

* autenticarFuncionario
* registrarNovoFuncionario
* guardarDataHoraAcesso

## 8. Plataformas

**Entidade**: Plataforma

**Atributos**:

* ID (Inteiro)
* nomePlataforma (String)
* email (String)
* senha (String)
* dataHoraCriacao (String)

**Métodos**:

* autenticarPlataforma
* registrarNovaPlataforma
* guardarDataHoraCriacao

# Relacionamentos

## Usuário x Plataforma de login:

**Relacionamento:** Um para Muitos e Muitos para Um

Exemplos:

Um usuário pode estar associado a vários logins e cadastro, mas um login em cada plataforma está associado a apenas um visitante.

## Monitoramento da Jornada do Usuário:

**Relacionamento:** Muitos para Um com Usuário

Exemplo:

Muitas interações de usuários podem estar associadas a um único usuário, ou seja, o usuário pode gerar várias informações durante sua jornada no site.

## Atualização do Conteúdo ou Sinalização de Melhorias:

**Relacionamento:** Muitas para Um com Usuário e Funcionários

Exemplo:

Muitas atualizações de conteúdo podem estar associadas a um único usuário e cada funcionário pode se basear nas interações e registros para solicitar melhorias.

## Idioma de Interação:

**Relacionamento:** Muitos para Um com Usuário

Exemplo:

Muitos idiomas de interação podem estar associados a um único usuário. Ele pode alterar o idioma a cada tipo de produto ou interação na página, desde seleção do texto, até ouvir ao suporte da tela falando.

## Segurança:

**Relacionamento:** Muitos para Um com Usuário

Exemplo:

Muitas configurações de segurança podem estar associadas a um único usuário, desde autenticação, até o momento de login, troca de senha e cadastro no primeiro acesso.

## Login\_Acesso:

**Relacionamento:** Muitos para Um com Usuário

Exemplo:

Muitos acessos podem estar associados a um único usuário.

## Banco de dados

**Relacionamento:** Um para Muito com Jornada usuário

Exemplo:

A cada tela de produto que o usuário passar, pode gerar uma ou muitas linhas de informações no banco de dados.

## Segurança x Login

**Relacionamento:** Muitos para Muitos

Exemplo:

Várias tentativas de login podem gerar novas conexões e validações/autenticação de acessos.