

Curso Superior de Desenvolvimento de Software Multiplataforma

Nome e RA dos integrantes

Adryann Theyllor Fernandes Pontes - 3011392413027

Davy Oliveira Ribeiro - 3011392413004

Matheus Augusto Santos Gueff - 3011392413006

Projeto Interdisciplinar

Engenharia de Software II

Desenvolvimento Web II

Banco de Dados Relacional

SaneaSP

Orientadores: Prof^a Cristiane Palomar Mercado

Prof^a Maria Janaína da Silva Ferreira

Prof^o Jones Artur Gonçalves

Prof^o Tiago Vanderlei de Arruda

Votorantim

Novembro, 2024

RESUMO

O projeto SaneaSP tem como objetivo criar um site para fortalecer a comunicação entre a população e os responsáveis pela área de saneamento em Sorocaba/Votorantim, contribuindo para duas ODS: Água Potável e Saneamento (ODS6) e Saúde e Bem-Estar (ODS3). O projeto permitirá que os cidadãos relatem problemas de forma estruturada e fácil e acompanhem as soluções para esses problemas, além de disponibilizar informações organizadas sobre doenças relacionadas e ações corretivas e outras notícias relacionadas à área de saneamento básico. Dessa forma, buscamos contribuir para a eficiência e transparência das iniciativas já existentes.

Sumário

1. DESCRIÇÃO DO PROJETO	1
1.1. Proposta do Software	1
1.2. Justificativa.....	1
1.3. Mapa do Site	2
1.4. Logomarca	2
2. REQUISITOS DO PROJETO.....	3
2.1. Levantamento de Requisitos	3
2.2. Requisitos Funcionais	3
2.3. Diagrama de Caso de Uso	5
2.4. Descrição do Caso de Uso.....	5
2.5. Requisitos Não Funcionais.....	12
3. PROJETO DO SOFTWARE	13
3.1. Tecnologias Utilizadas	13
3.2. Modelo de dados	14
3.2.1 Modelo Conceitual	14
3.2.2 Modelo Lógico.....	15
3.3. Diagrama de Classe.....	15
3.4. Diagrama de Atividades	16
3.5. Diagrama de Sequência.....	18
3.6. Interfaces com o usuário	18
4. ESTRATÉGIA DE TESTES	35
5. IMPLANTAÇÃO	36

LISTA DE FIGURAS

Figura 1 - Mapa mental do Site	2
Figura 2 - Logomarca do projeto	3
Figura 3 - Caso de Uso	5
Figura 4 - Diagrama de Entidade Relacionamento (DER)	14
Figura 5 - Modelo Lógico	15
Figura 6 - Diagrama de Classes.....	16
Figura 7 - Diagrama de Atividade "Adicionar Reclamação"	17
Figura 8 - Diagrama de Atividades "Comentários"	18
Figura 9 - Diagrama de Sequência "Reclamação"	18
Figura 10 – Cabeçalho e barra de navegação	19
Figura 11 – Rodapé do site	19
Figura 12 – Página Inicial dos Usuários	19
Figura 13 – Continuação (Página Inicial dos Usuários)	20
Figura 14 – Página de Login	20
Figura 15 – Página de Cadastro	21
Figura 16 – Continuação (Página de Cadastro)	21
Figura 17 - Aviso de Cadastro realizado com sucesso.....	22
Figura 18 - Aviso de login realizado com sucesso	22
Figura 19 – Barra de navegação após o login.....	22
Figura 20 - Barra de navegação após o login.....	23
Figura 21 – Editar Perfil	23
Figura 22 – Continuação (Editar Perfil)	23
Figura 23 – Página Inicial do Administrador	24
Figura 24 – Página de Doenças.....	24
Figura 25 – Continuação (Página de Doenças).....	25
Figura 26 – Página de Descrição da Doença	25

Figura 27 – Continuação (Página da Descrição da Doença)	26
Figura 28 – Card para nenhum resultado encontrado	26
Figura 29 – Página de Notícias	27
Figura 30 - Continuação (Página de Notícias).....	27
Figura 31 – Página de Reclamações	28
Figura 32 – Continuação (Página de Reclamações)	28
Figura 33 – Continuação (Página de Reclamações)	28
Figura 34 – Página de Descrição da Reclamação.....	29
Figura 35 – Página de Comentários.....	29
Figura 36 - Continuação (Página de Comentários)	30
Figura 37 - Formulário de Cadastro de Reclamação	30
Figura 38 - Continuação (Formulário de Cadastro de Reclamação).....	31
Figura 39 - Pré Visualização de Imagem	31
Figura 40 - Formulário de Cadastro de Notícia	32
Figura 41 - Continuação (Formulário de Cadastro de Notícias).....	32
Figura 42 - Inserção de tag e fonte	32
Figura 43 - Formulário de Cadastro de Doenças.....	33
Figura 44 – Continuação (Formulário de Cadastro de Doenças).....	33
Figura 45 - Continuação (Formulário de Cadastro de Doenças)	33
Figura 46 – Página Sobre Nós	34
Figura 47 - Continuação (Página Sobre Nós).....	34
Figura 48 – Página dos Responsáveis.....	34
Figura 49 – Continuação (Página dos Responsáveis).....	35

LISTA DE QUADROS

Quadro 1 - Requisitos funcionais	4
Quadro 2 - Caso de uso "Cadastro"	5
Quadro 3 - Caso de uso "Login"	6
Quadro 4 - Caso de uso "Login com redes sociais"	7
Quadro 5 - Caso de uso "Reclamação"	7
Quadro 6 - Caso de uso "Comentário"	8
Quadro 7 - Caso de uso "Gerar log"	9
Quadro 8 - Caso de uso "Informações do site"	9
Quadro 9 - Caso de Uso "Administrar Doenças"	10
Quadro 10 - Caso de uso "Administrar Notícias"	10
Quadro 11 - Caso de uso "Configurar perfil"	11
Quadro 12 - Requisitos não funcionais	12

1. DESCRIÇÃO DO PROJETO

1.1. Proposta do Software

O presente projeto tem como objetivo abranger duas ¹ODSs: Água Potável e Saneamento (ODS6) e Saúde e Bem-Estar (ODS3), por meio de um website. A função do site SaneaSP é auxiliar a prefeitura ou o órgão responsável pelo saneamento básico e administração da água da cidade, disponibilizando de forma rápida e organizada uma série de comentários e denúncias de problemas que estão ocorrendo na região, servindo como um centro de informações para os responsáveis.

Além disso, o site disponibilizará informações aos usuários, permitindo o acesso aos dados sobre saneamento e água potável de forma rápida e fácil, ajudando os usuários a se prevenirem de doenças e a ficarem atentos às notícias recentes sobre esse assunto.

Dessa forma, o projeto prevê reduzir a invisibilidade dos problemas de saneamento na sua cidade, contribuindo tanto para os responsáveis quanto para os cidadãos do local.

1.2. Justificativa

Foram escolhidas as ODSs seis e três (Água Potável e Saneamento, e Saúde e Bem-Estar, respectivamente), pois operam uma relação de causa e consequência entre si, já que populações com saneamento básico e água potável estão protegidas de uma ampla gama de doenças.

Todavia, problemas de saneamento e esgoto, como as fossas a céu aberto e canos de água estourados não recebem a devida atenção dos órgãos responsáveis. Mesmo com a infraestrutura para resolver tais problemas, muitas vezes não há a percepção de que esses problemas sequer estão acontecendo.

¹ ODS

Além disso, a desinformação da população em relação as consequências da falta de saneamento básico, como as doenças que ela dissemina, dificulta a comunicação entre a governança e a população, pois ambos não sabem que existe um problema a ser resolvido.

Somado a isso, a má divulgação dificulta que a população saiba se a prefeitura está se mobilizando para resolver estes problemas, o que a desestimula a denunciá-los.

Portanto, haja vista esses obstáculos, foi analisada uma chance de colaborar para a redução desses problemas, por meio de um site focado justamente em tornar a manutenção do saneamento básico e as atitudes da governança mais visíveis e mostrar os problemas do cotidiano, informar, e divulgar notícias para a população.

1.3. Mapa do Site

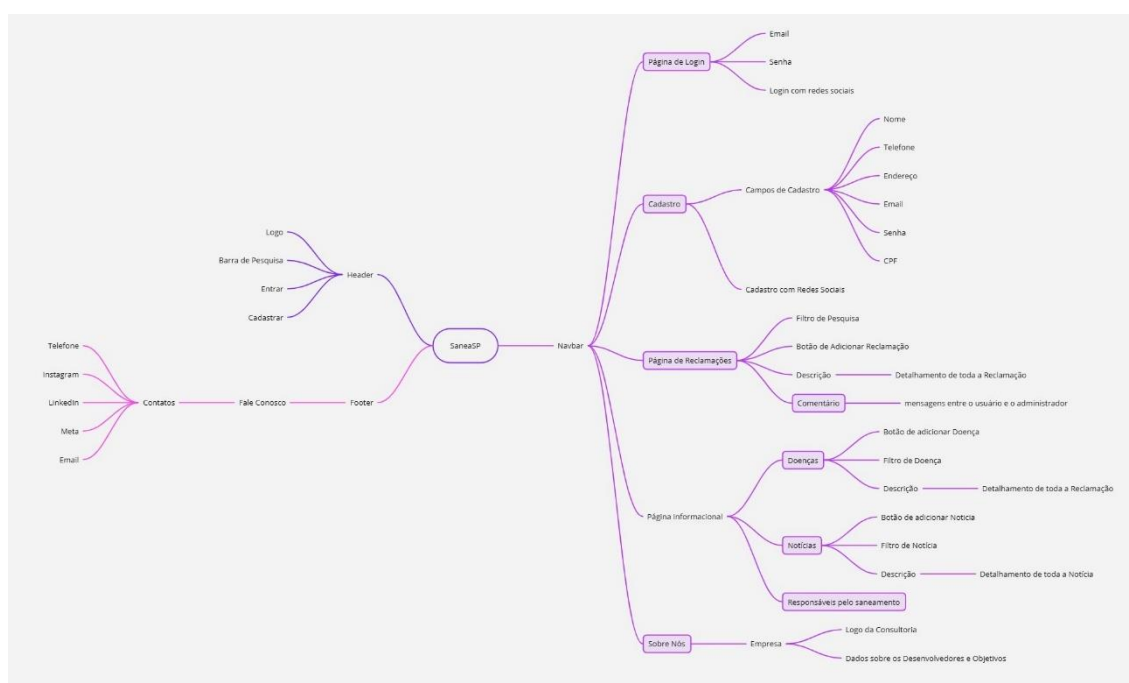


Figura 1 - Mapa mental do Site

1.4. Logomarca



Figura 2 - Logomarca do projeto

2. REQUISITOS DO PROJETO

2.1. Levantamento de Requisitos

Para o levantamento de requisitos, foi observado como é o cotidiano do saneamento básico em Votorantim que é a cidade alvo do projeto, então foi percebida a necessidade da criação de um site que reúna informações e que torne a comunicação da população mais fácil.

Uma das ferramentas que foi utilizada como base é o site Reclame Aqui, que dispõe uma comunicação entre o usuário e o serviço que ele avalia, assim como o site deste projeto, que busca criar essa facilidade para os responsáveis pelo saneamento da cidade, com o sistema de comentários e respostas.

2.2. Requisitos Funcionais

Quadro 1 - Requisitos funcionais

Nº Requisito	Nome	Descrição
RF001	Login	Cadastro e Login de usuários ou administradores do site.
RF002	Cadastro de Reclamações	Página de adição de reclamações.
RF003	Gerenciamento de Reclamações	Funções para que o usuário pode modificar em reclamações criadas por ele ou para administradores para todas as reclamações.
RF004	Cadastro de Usuário	Página de Adição de usuário.
RF005	Pesquisa de Reclamação	Pesquisa o título da reclamação no banco de dados.
RF006	Upload de Imagens	O usuário poderá enviar a imagem do problema que encontrou.
RF007	Exibir Doenças	Informações sobre as doenças relacionadas à falta de saneamento básico.
RF008	Registro com redes sociais	O usuário poderá fazer seu registro utilizando as suas redes sociais, utilizando sua conta google, LinkedIn ou facebook.
RF009	Exibir Reclamações	Exibe as reclamações feitas pelos usuários para que todos outros possam visualizar.
RF010	Filtragem de pesquisa	Filtros existentes na página de reclamações que filtraram as reclamações a serem exibidas através de tipos e bairros.
RF011	Exibição completa da reclamação	Quando o usuário clicar numa reclamação ela se expandirá para aparecer todo o conteúdo.
RF012	Barra de Navegação	Barra para o acesso rápido de outras áreas do site.
RF013	Exibir órgãos responsáveis	Haverá no footer um link direcionando o usuário para a página de órgãos responsáveis.
RF014	Exibir doenças relacionadas	Haverá no footer um link direcionando o usuário para a página de doenças relacionadas.
RF015	Gerar log	Quando o usuário/administrador editar uma reclamação ou comentário, será gerado um log

		para guardar as informações alteradas.
RF016	Geração de pontuação	Quando o usuário fizer sua reclamação, será gerada uma pontuação baseada nas informações fornecidas

2.3. Diagrama de Caso de Uso

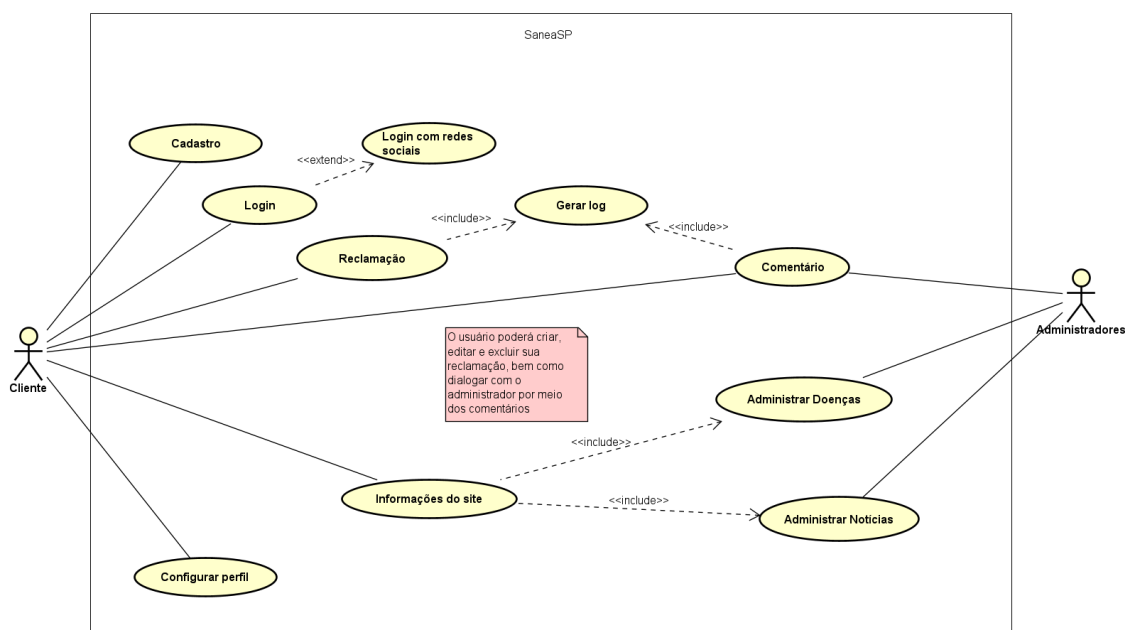


Figura 3 - Caso de Uso

2.4. Descrição do Caso de Uso

Quadro 2 - Caso de uso "Cadastro"

Caso de uso	Cadastrar Usuário
Ator Principal	Usuário
Ator Secundário	-
Pré-condições	Usuário não possuir login ou querer criar uma nova conta
Pós-condições	O usuário possui seus dados cadastrados no sistema, possibilitando efetuar seu login
Fluxo Principal	
Ações do Ator	Ações do Sistema

CADASTRAR	
1. Acessar página de cadastro	
	2. Sistema exibe o formulário
3. Usuário preenche o formulário com seus dados (Nome, email, senha, telefone, cpf e endereço)	
	4. Sistema valida se campos obrigatórios estão preenchidos, se o email já existe no sistema e se é um email válido, se o CEP é válido, e se as senhas coincidem na confirmação da senha
	5. Sistema guarda as informações no banco de dados
	6. Retorna uma mensagem de cadastro realizado com sucesso
	7. O usuário é redirecionado para a página de login.

Quadro 3 - Caso de uso "Login"

Caso de uso	Login de usuário	
Ator Principal	Usuário	
Ator Secundário	-	
Pré-condições	Usuário possuir seus dados já cadastrados no sistema	
Pós-condições	O usuário pode realizar reclamações no site	
Fluxo Principal		
Ações do Ator		Ações do Sistema
LOGIN		
1. Acessar página de Login		
		2. Sistema exibe o formulário
3. Usuário preenche o formulário com seus dados (email e senha)		
		4. Sistema valida se todos campos estão preenchidos, se o email existe no sistema e se a senha é válida para esse usuário.
		5. Retorna uma mensagem se o login foi efetuado com sucesso

	6. O usuário é redirecionado para a página principal do site.
--	---

Quadro 4 - Caso de uso "Login com redes sociais"

Caso de uso	Login com redes sociais		
Ator Principal	Usuário		
Ator Secundário	-		
Pré-condições	O usuário deve possuir um cadastro em alguma rede social		
Pós-condições	O usuário pode realizar reclamações no site		
Fluxo Principal			
Ações do Ator		Ações do Sistema	
LOGIN COM REDES SOCIAIS			
1. Acessar página de Login			
		2. Sistema exibe o formulário	
3. Usuário escolhe a rede social para efetuar login			
		4. Sistema redireciona o usuário para a página de login da rede social	
		5. Retorna uma mensagem se o login foi efetuado com sucesso	
		6. Redireciona o usuário para a página principal	

Quadro 5 - Caso de uso "Reclamação"

Caso de uso	Reclamações	
Ator Principal	Usuário	
Ator Secundário	-	
Pré-condições	O usuário deve estar com seu login	
Pós-condições		
Fluxo Principal		
Ações do Ator		Ações do Sistema
RECLAMAÇÃO		
1. Acessar página de adicionar reclamações		
		2. Sistema exibe o formulário

3. O usuário preenche os dados do formulário com os campos necessários	
	4. Sistema valida se os campos obrigatórios foram preenchidos e se há um endereço válido.
	5. Sistema valida se o usuário enviou uma imagem, e se houver, valida o tamanho e tipo do arquivo enviado
	6. O sistema gera pontuação baseada na informação fornecida e na confiabilidade da reclamação
	7. Sistema guarda as informações no banco de dados
	8. Retorna uma mensagem se a reclamação foi cadastrada com sucesso
	9. Redireciona o usuário para a página de reclamações

Quadro 6 - Caso de uso "Comentário"

Caso de uso	Comentário
Ator Principal	Usuário
Ator Secundário	-
Pré-condições	Necessita de uma reclamação
Pós-condições	
Fluxo Principal	
Ações do Ator	Ações do Sistema
RESPOSTA	
1. Acessar página de comentários	
	2. Sistema exibe o histórico das mensagens e uma caixa de texto para digitar seu comentário
3. O usuário preenche a caixa de texto com seu comentário	
	4. Sistema valida se o usuário preencheu a caixa de texto e se foram digitadas palavras ofensivas
	5. Sistema grava as informações no

	banco de dados
	6. Sistema atualiza a página, com os dados do comentário do usuário preenchidos

Quadro 7 - Caso de uso "Gerar log"

Caso de uso	Gerar log
Ator Principal	Usuário
Ator Secundário	-
Pré-condições	Reclamação ou comentário ser editado
Pós-condições	
Fluxo Principal	
Ações do Ator	Ações do Sistema
GERAR LOG	
1. O ator edita um comentário ou uma reclamação criada	
	2. O sistema utiliza um trigger para registrar o conteúdo antigo e o novo em linhas da tabela de logs

Quadro 8 - Caso de uso "Informações do site"

Caso de uso	Informações do site
Ator Principal	Usuário
Ator Secundário	-
Pré-condições	
Pós-condições	
Fluxo Principal	
Ações do Ator	Ações do Sistema
INFORMAÇÕES DO SITE	
1. Acessar páginas informativas	
	2. Sistema exibe as informações em ordem de atualidade
3. O usuário pesquisa por tags para filtragem das informações ou por nome	
	4. O sistema valida se existem informações com a tag especificada ou se existe alguma

	informação com o nome especificado
--	------------------------------------

Quadro 9 - Caso de Uso "Administrar Doenças"

Caso de uso	Administrar Doenças		
Ator Principal	Administrador		
Ator Secundário	-		
Pré-condições	Possuir login de administrador		
Pós-condições			
Fluxo Principal			
Ações do Ator		Ações do Sistema	
ADMINISTRAR DOENÇAS			
1. Acessar página de cadastro de informações			
		2. Sistema exibe o formulário para o cadastro da informação	
3. O administrador preenche os campos do formulário			
		4. O sistema valida se os dados das informações são válidos e se não são duplicados, se houver imagem, valida se o tamanho e tipo são válidos	
		5. Sistema guarda os dados no banco de dados.	
		6. Sistema informa se foi efetuado com sucesso e retorna uma mensagem.	

Quadro 10 - Caso de uso "Administrar Notícias"

Caso de uso	Administrar notícias	
Ator Principal	Administrador	
Ator Secundário	-	
Pré-condições	Possuir login de administrador	
Pós-condições		
Fluxo Principal		
Ações do Ator		Ações do Sistema
ADMINISTRAR NOTÍCIAS		

1. Acessar página de cadastro de informações	
	2. Sistema exibe o formulário para o cadastro da informação
3. O administrador preenche os campos do formulário	
	4. O sistema valida se os dados das informações são válidos e se não são duplicados, valida se o tamanho e tipo são válidos
	5. Sistema guarda as informações no banco de dados.
	6. Sistema informa se foi efetuado com sucesso e retorna uma mensagem.

Quadro 11 - Caso de uso "Configurar perfil"

Caso de uso	Configurar perfil		
Ator Principal	Usuário		
Ator Secundário	-		
Pré-condições	Possuir um login		
Pós-condições			
Fluxo Principal			
Ações do Ator		Ações do Sistema	
CONFIGURAR PERFIL			
1. Acessar página de alteração de perfil			
		2. Sistema exibe o formulário com valores preenchidos pelo banco	
3. O usuário preenche o formulário e clica em salvar			
		4. O sistema valida se houve alguma alteração no perfil e se o email é válido e não duplicado. Caso o preenchimento tenha sucesso, o sistema exibe um formulário para verificação de senha do usuário	
5. O usuário digita sua senha			
		6. O sistema valida se a senha digitada corresponde com a	

	senha registrada para aquele usuário
	7. Sistema guarda as informações no banco de dados
	8. Sistema informa que foi alterado com sucesso e retorna uma mensagem.
	9. O sistema redireciona o usuário para a tela principal

2.5. Requisitos Não Funcionais

Quadro 12 - Requisitos não funcionais

Nº Requisito	Nome	Descrição
RNF001	Acesso à Internet	Os usuários terão que ter acesso para acessar o site através da internet
RNF002	Banco de Dados	Estrutura na nuvem na qual os cadastros e as reclamações serão armazenados para utilização do site.
RNF003	Responsividade	O site terá responsividade para ter um bom acesso em diferentes plataformas
RNF004	Hardwares mínimos	Será necessário que o usuário tenha pelo menos um computador ou dispositivo que acesse a internet e abra os navegadores
RNF005	Servidor	É necessário para o funcionamento do site que esteja hospedado num servidor com acesso de banco de dados e todos os recursos web que o site necessite.
RNF006	Acessibilidade	O site deverá ter acessibilidade para daltônicos, cegos, pessoas com visão debilitada e gerais.
RNF007	Animações do Site	Animações por meio do CSS e Javascript
RNF008	Notificações	Notificar quando acontecer um erro no banco de dados, ou quando um processo for concluído corretamente com

		uma função 'alert' em Javascript.
RNF009	link para a página de reclamações	Haverá no footer um link direcionando o usuário para a página de reclamações.

3. PROJETO DO SOFTWARE

3.1. Tecnologias Utilizadas

Bootstrap:

²É um framework front-end que fornece estruturas de CSS para a criação de sites e aplicações responsivas de forma rápida e simples. Além disso, pode lidar com sites de desktop e páginas de dispositivos móveis da mesma forma

Astah:

³Ferramenta de diagramação para indivíduos e equipes de todos os tamanhos. As ferramentas do Astah permitem criar uma grande variedade de diagramas e oferecem preços flexíveis adequados às suas necessidades.

Angular:

⁴Angular é um framework de código aberto desenvolvido pelo Google para a criação de aplicativos dinâmicos e interativos da web.

Ele se baseia em TypeScript, uma linguagem superset do JavaScript, e utiliza uma abordagem orientada a componentes para a construção de interfaces. O Angular oferece uma estrutura sólida para desenvolvimento, incluindo suporte a gerenciamento de estado, roteamento, validação de formulários e muito mais.

² <https://www.alura.com.br/artigos/bootstrap>

³ <https://astah.net/pt/>

⁴ <https://www.alura.com.br/artigos/angular-js>

SweetAlert2:

⁵O SweetAlert é uma API — Application Programming Interface — utilizada para estilizar e adicionar funcionalidades às caixas de diálogos em aplicações web. Ele oferece uma série de recursos que tornam a comunicação entre a página e a pessoa usuária muito mais funcional e elegante.

3.2. Modelo de dados

3.2.1 Modelo Conceitual

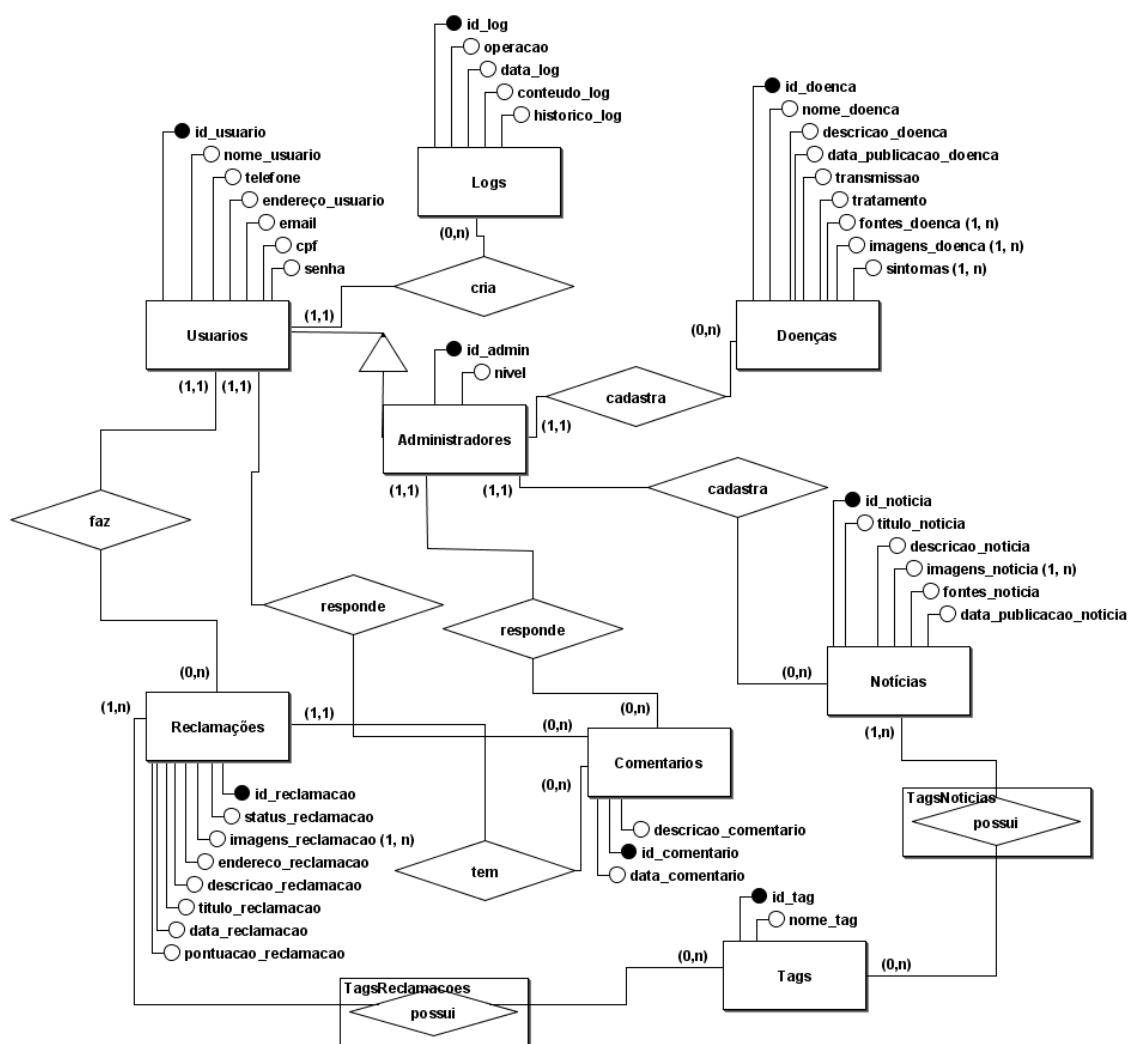


Figura 4 - Diagrama de Entidade Relacionamento (DER)

⁵ <https://blog.betrybe.com/desenvolvimento-web/sweetalert/>

3.2.2 Modelo Lógico

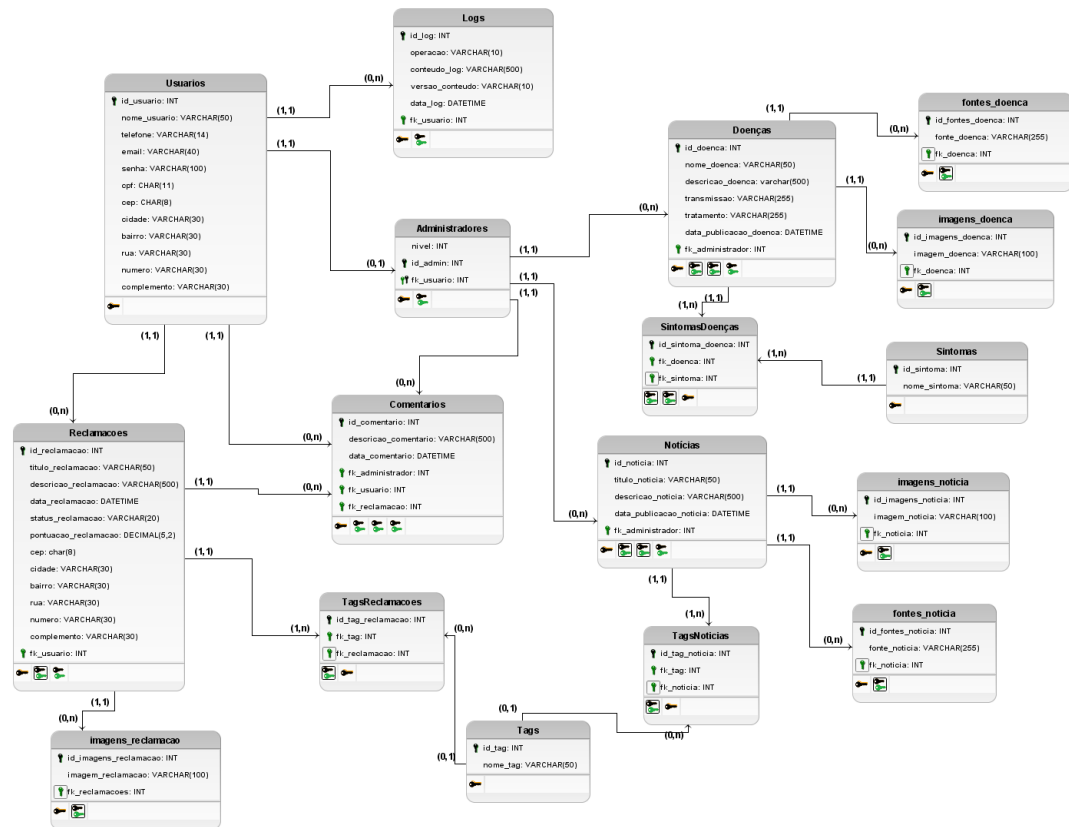


Figura 5 - Modelo Lógico

3.3. Diagrama de Classe

3.4. Diagrama de Atividades

3.4. Diagrama de Atividades

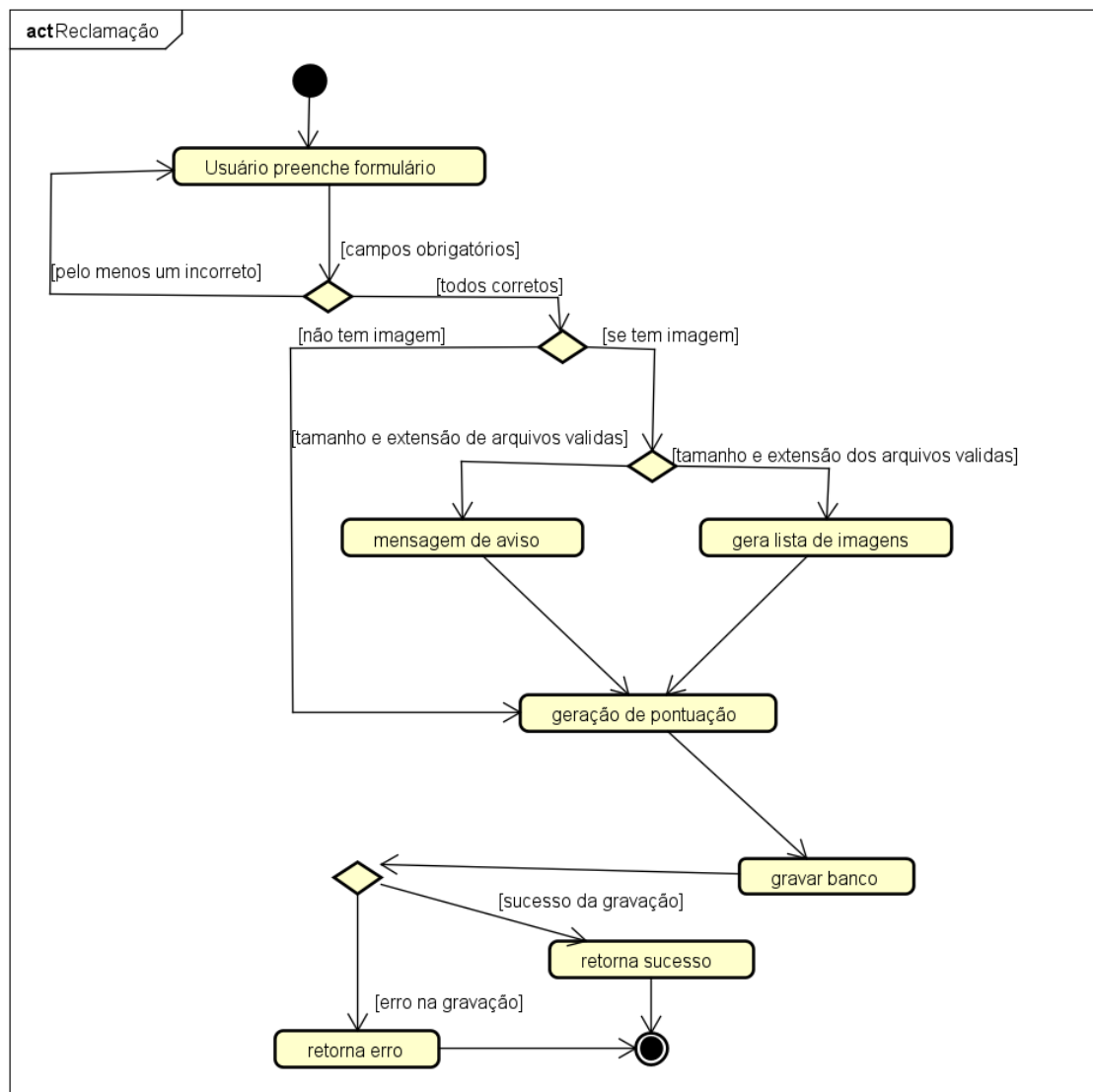


Figura 7 - Diagrama de Atividade "Adicionar Reclamação"

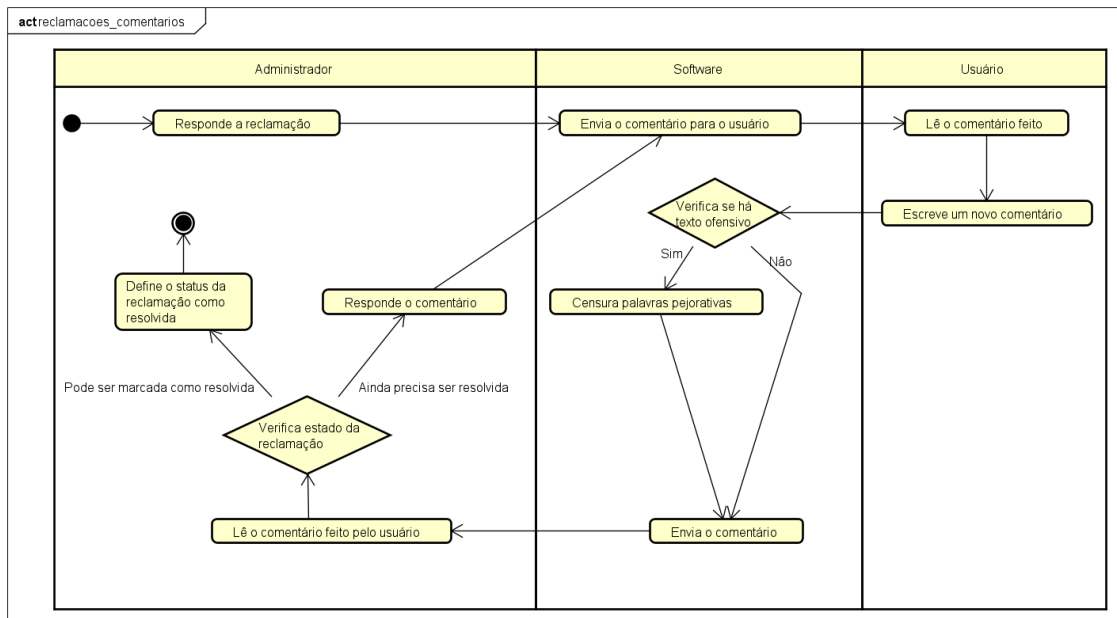


Figura 8 - Diagrama de Atividades "Comentários"

3.5. Diagrama de Sequência

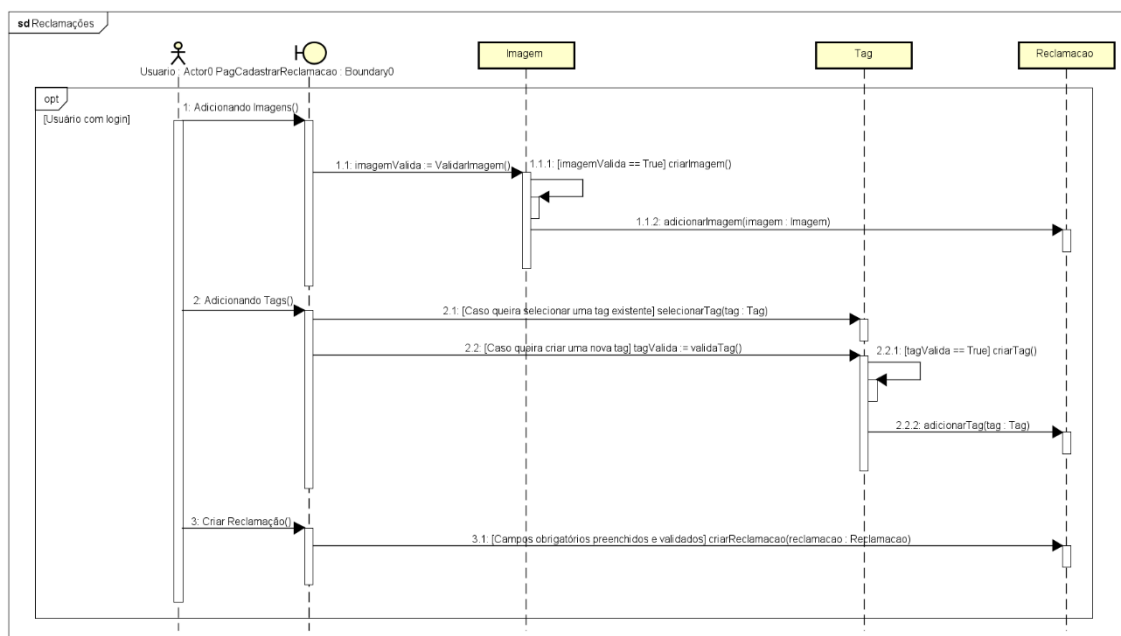


Figura 9 - Diagrama de Sequência "Reclamação"

3.6. Interfaces com o usuário

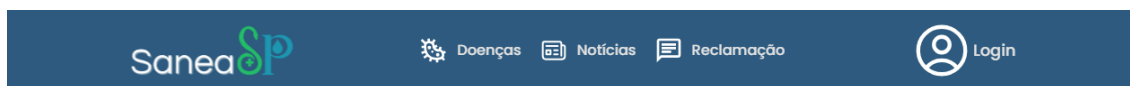


Figura 10 – Cabeçalho e barra de navegação

Cabeçalho e Barra de Navegação: Possui um link para a página inicial (clcando na logo do site), os links para as páginas principais e para a página de login

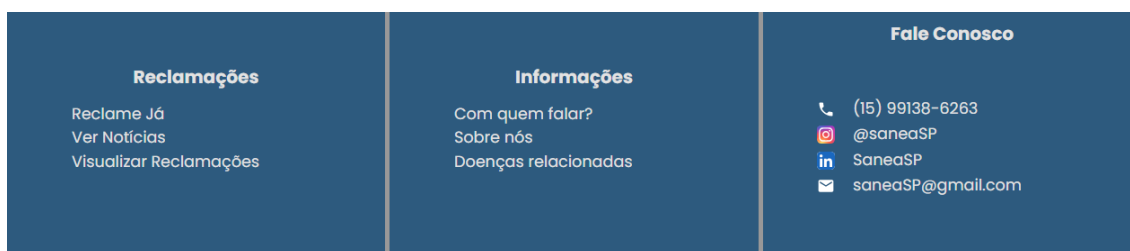


Figura 11 – Rodapé do site

Rodapé: Links para as páginas principais e para os meios de contato com a consultoria

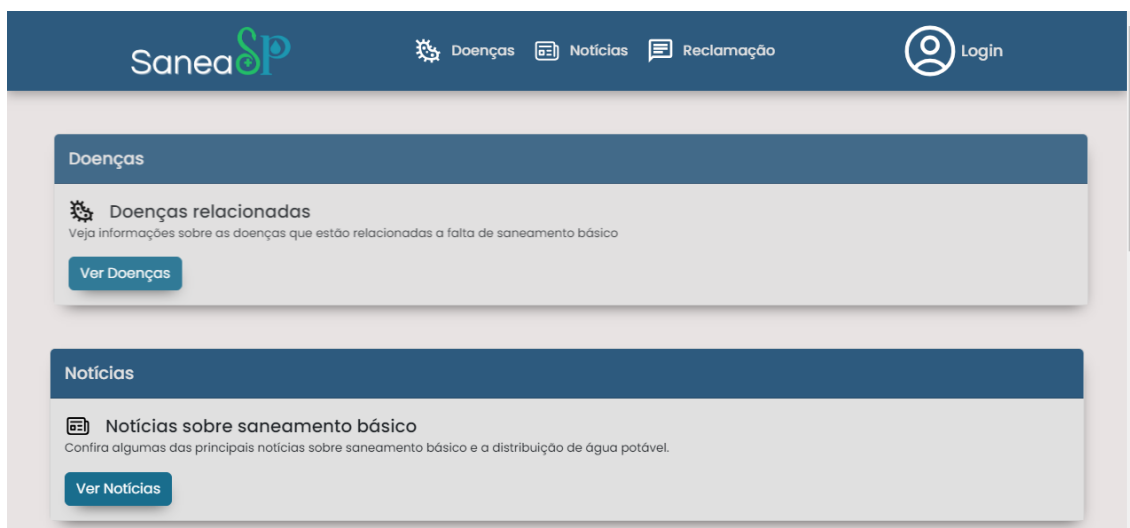


Figura 12 – Página Inicial dos Usuários



Figura 13 – Continuação (Página Inicial dos Usuários)

Página inicial do site para usuários comuns: Nela temos quatro blocos de conteúdo – Doenças, Notícias, Reclamações e Responsáveis – cada um com uma breve descrição e links para demais páginas.

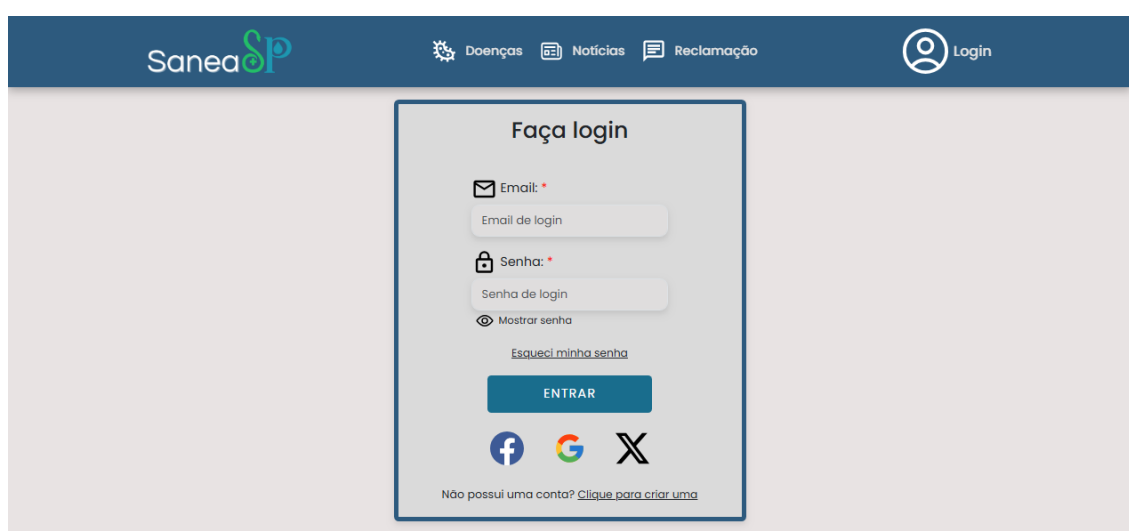


Figura 14 – Página de Login

Página Login: Clicando no botão Login, a página será direcionada para o usuário poder fazer o login, colocando email, senha ou logar pelas redes sociais, caso tenha cadastro. Se não tiver poderá clicar em “clique para criar uma”

Figura 15 – Página de Cadastro

Figura 16 – Continuação (Página de Cadastro)

Página de Cadastro: página para cadastrar um novo usuário. Sendo os campos obrigatórios os de asterisco em vermelho (*). Caso já possua um login o usuário pode clicar em “Faça Login” ao final do formulário.

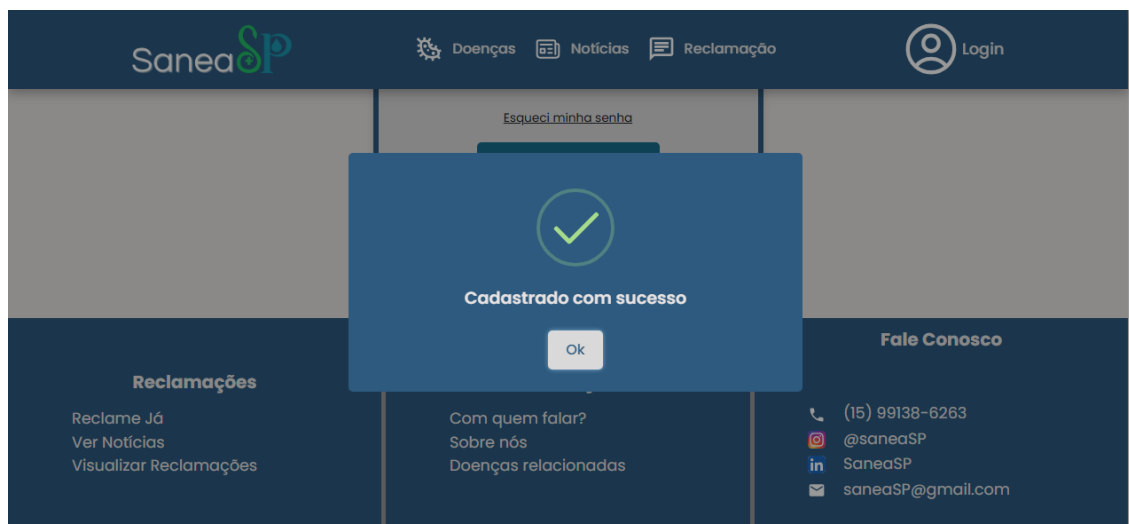


Figura 17 - Aviso de Cadastro realizado com sucesso

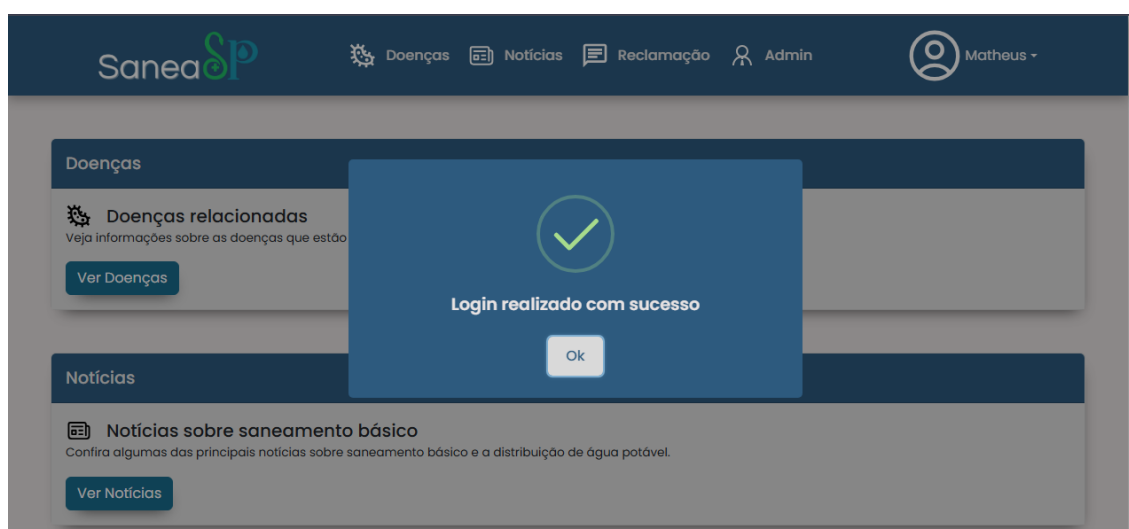


Figura 18 - Aviso de login realizado com sucesso

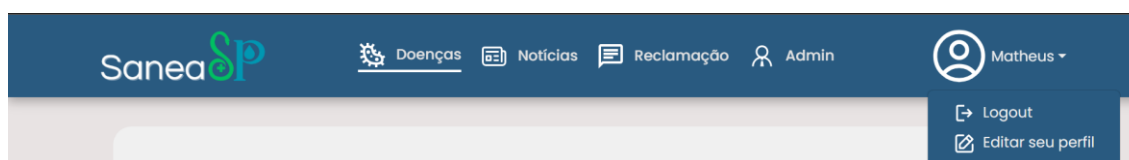


Figura 19 – Barra de navegação após o login

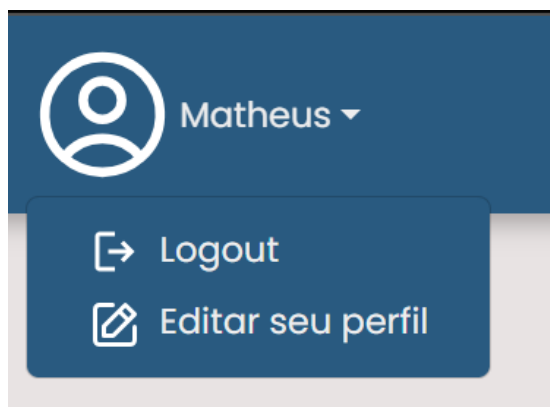


Figura 20 - Barra de navegação após o login

Barra de navegação após login: A barra de navegação fica semelhante ao a barra de navegação sem o usuário estar logado, exceto pelo botão de login que agora mostra o nome do usuário e quando pressionado aparece duas opções: Logout e “Editar seu Perfil”

Figura 21 – Editar Perfil

Figura 22 – Continuação (Editar Perfil)

Editar Perfil: Editar perfil puxa todos os dados do usuário logado e mostra eles para que ele possa alterar.



Figura 23 – Página Inicial do Administrador

Página Inicial Admin: mostra todas as páginas o administrador tem acesso



Figura 24 – Página de Doenças

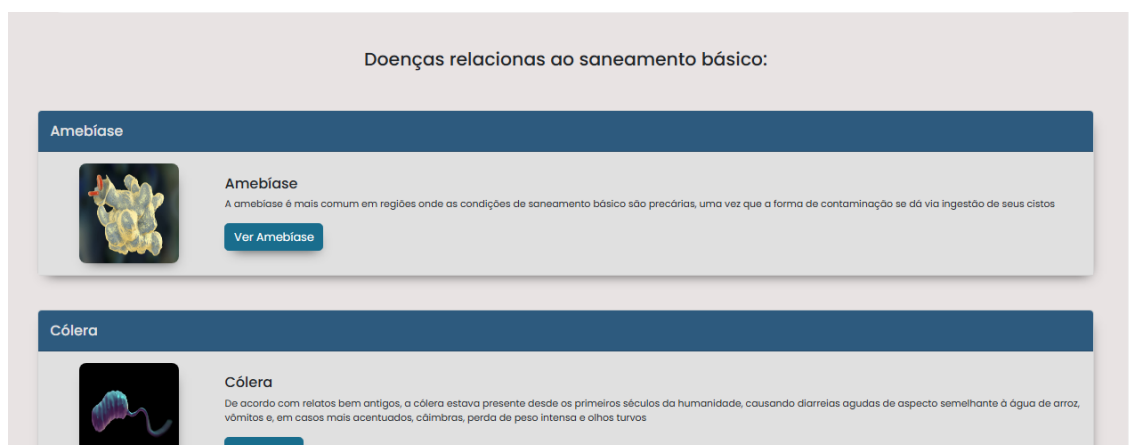


Figura 25 – Continuação (Página de Doenças)

Página Doenças: página que mostra todas as doenças cadastradas. Caso o usuário se interesse por uma doença, pode clicar no botão e ver a descrição sobre a doença

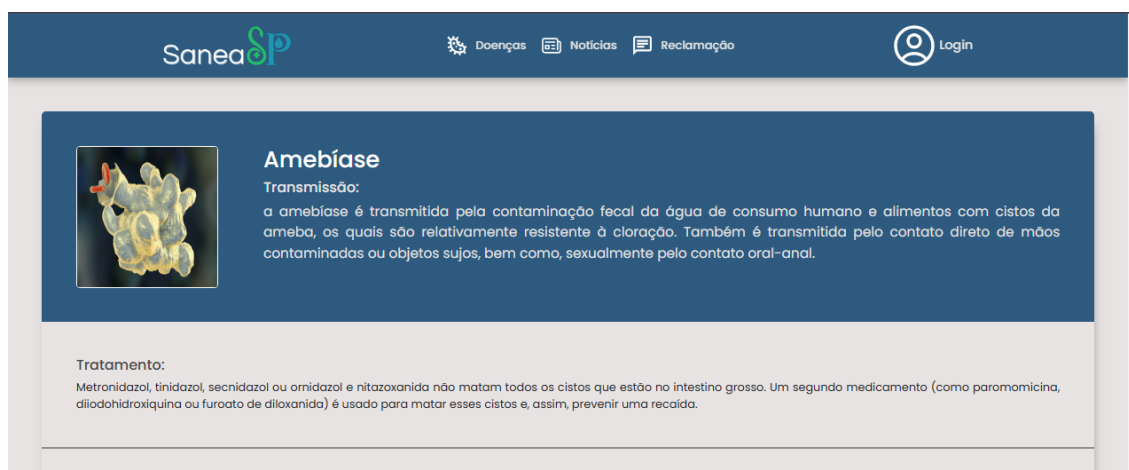


Figura 26 – Página de Descrição da Doença



Figura 27 – Continuação (Página da Descrição da Doença)

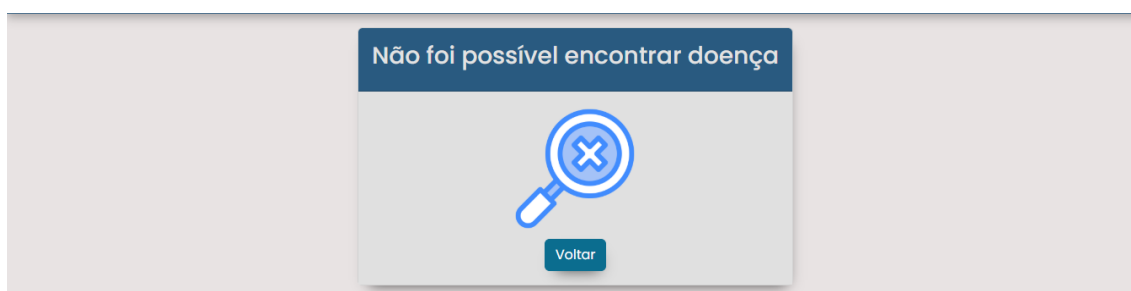


Figura 28 – Card para nenhum resultado encontrado

Página de Descrição da Doença: descrição detalhada da doença, tendo o modo de transmissão, tratamento e os sintomas, bem como o link para o artigo sobre a doença.



Figura 29 – Página de Notícias



Figura 30 - Continuação (Página de Notícias)

Página de Notícias: página que mostra todas as notícias cadastradas, podendo filtrar por tags atribuídas (como a tag “Poluição” na figura). Caso o usuário queira ver uma notícia, pode clicar no botão e ver os detalhes sobre ela.



Figura 31 – Página de Reclamações

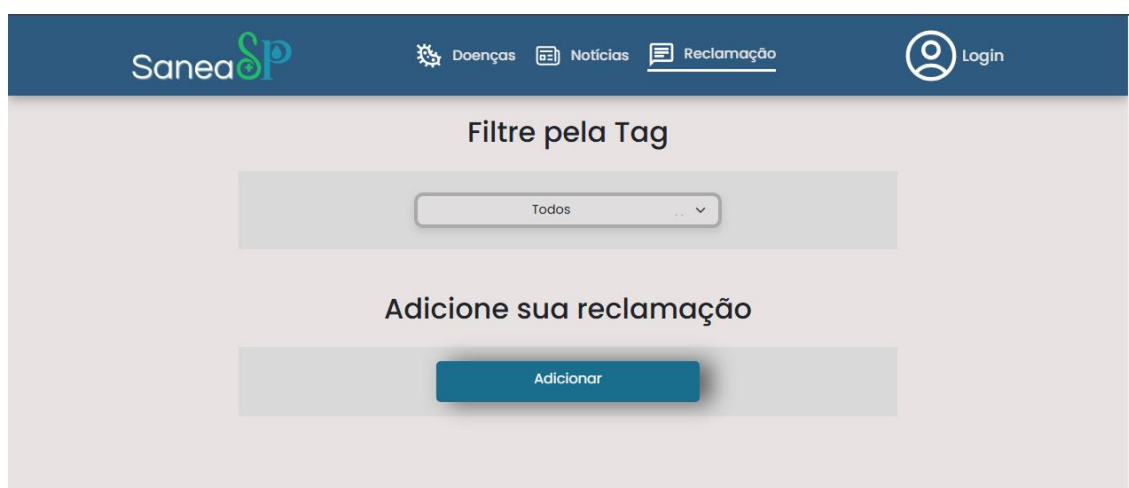


Figura 32 – Continuação (Página de Reclamações)



Figura 33 – Continuação (Página de Reclamações)

Página de Reclamações: página que mostra todas as reclamações cadastradas, podendo filtrar por tags atribuídas. Caso o usuário queira ver uma reclamação, pode clicar no botão e ver os detalhes e os comentários sobre ela. Um usuário com login pode criar uma nova reclamação no botão em “Adicione sua Reclamação”



Figura 34 – Página de Descrição da Reclamação

Página de Descrição da Reclamação: o usuário pode voltar para a página de reclamações ou ir para a sessão de comentários no botão de “Ir para Comentários”



Figura 35 – Página de Comentários

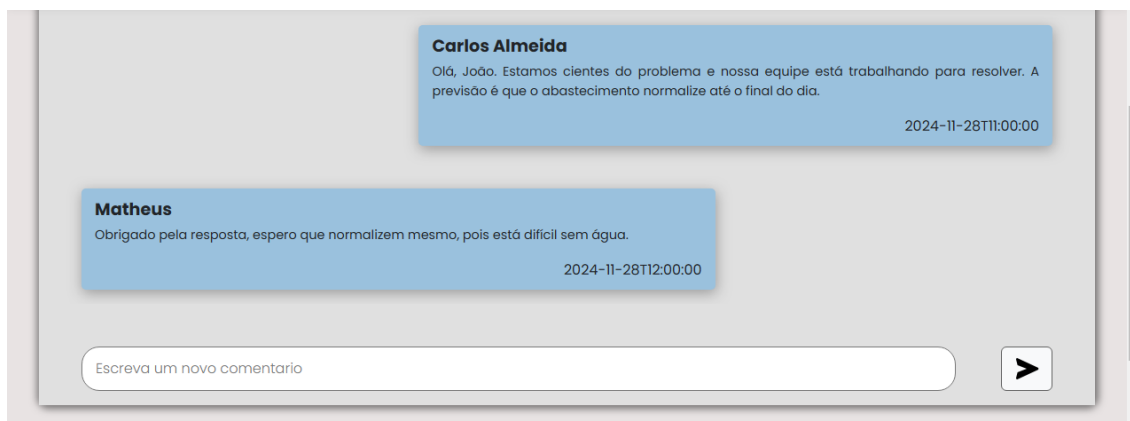


Figura 36 - Continuação (Página de Comentários)

Página de Comentários: nessa página o usuário pode visualizar os comentários feitos sobre a reclamação e adicionar seu próprio comentário ao final da página

Figura 37 - Formulário de Cadastro de Reclamação

Tags: Nenhum

Imagem: Escolher arquivo Nenhum arquivo escolhido

Pré-visualização

VOLTAR CADASTRAR

Reclamações	Informações	Fale Conosco
Reclame Já Ver Notícias Visualizar Reclamações	Com quem falar? Sobre nós Doenças relacionadas	<p>(15) 99138-6263</p> <p>@saneaSP</p> <p>SaneaSP</p> <p>saneaSP@gmail.com</p>

Figura 38 - Continuação (Formulário de Cadastro de Reclamação)

Escolher arquivo user1.jpg

Pré-visualização

VOLTAR CADASTRAR

Figura 39 - Pré Visualização de Imagem

Formulário de Reclamação: o usuário preenche os campos para cadastrar sua reclamação, campos de imagem e tags adicionam mais pontuação à reclamação que a torna mais visível para os administradores que devem responder as reclamações. Ao carregar uma imagem, ela fica disponível na pré visualização e pode ser retirada ao clicar nela

SaneaSP Doenças Notícias Reclamação Admin Matheus

Cadastre uma nova notícia

Título da notícia: *

Descrição da Notícia: *

Tag:

Adicionar tag

Adicione tags:

Figura 40 - Formulário de Cadastro de Notícia

Fonte:

Adicionar fonte

Adicione fontes:

Imagem:
 Nenhum arquivo escolhido

Pré-visualização

VOLTAR **CADASTRAR**

Figura 41 - Continuação (Formulário de Cadastro de Notícias)

Tag:

Adicionar tag

Tags adicionadas:

- Redução da Poluição

Fonte:

Adicionar fonte

Fontes adicionadas:

- <https://www.folha.uol.com.br/folha-topicos/saneamento/>

Figura 42 - Inserção de tag e fonte

Formulário de Notícias: o administrador preenche o formulário com dados da notícia, podendo adicionar fontes e tags que não são obrigatórios, ao carregar

uma imagem ela aparece na pré-visualização. Quando um usuário digitar uma tag ou fonte e apertar o botão (ou pressionar enter), ficará visível as tags/fontes inseridas, o usuário deve clicar sobre ela para remover da lista.

Figura 43 - Formulário de Cadastro de Doenças

Figura 44 – Continuação (Formulário de Cadastro de Doenças)

Figura 45 - Continuação (Formulário de Cadastro de Doenças)

Formulário de Doenças: o administrador preenche o formulário com dados da doença, o preenchimento dos campos “Sintomas” e “Fontes” ocorre assim como no formulário de notícias.



Figura 46 – Página Sobre Nós

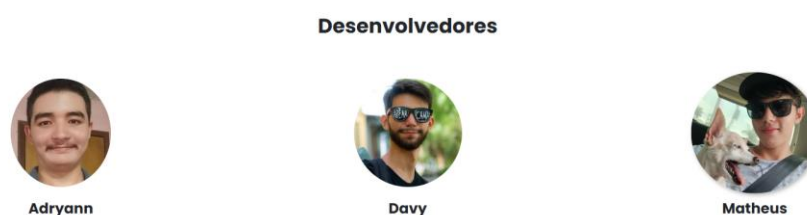


Figura 47 - Continuação (Página Sobre Nós)

Página Sobre Nós: Página que explica a origem da empresa, os objetivos e os desenvolvedores dela



Figura 48 – Página dos Responsáveis

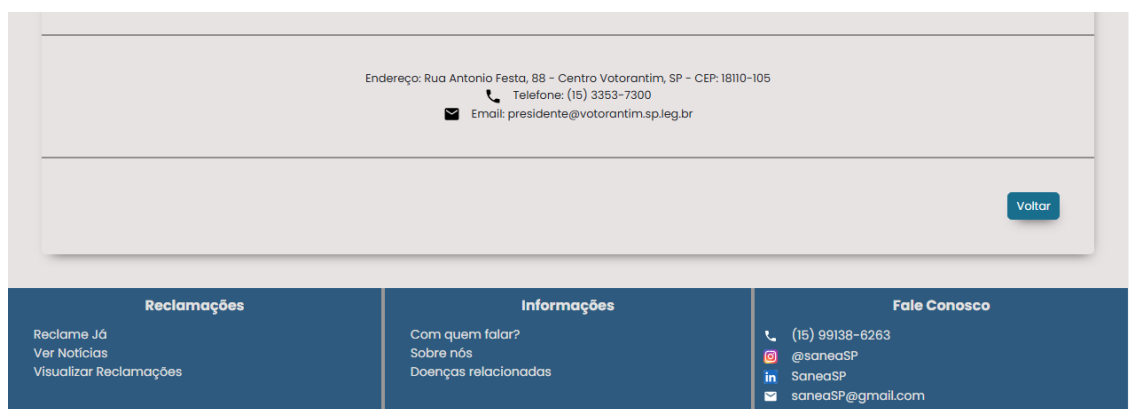


Figura 49 – Continuação (Página dos Responsáveis)

Página Responsáveis: página para exibição de informações sobre os responsáveis pelo saneamento da cidade

4. ESTRATÉGIA DE TESTES

Os testes a serem realizados no site consistem no conteúdo, na experiência dos usuários, na compatibilidade com diferentes navegadores no nosso sistema. Os testes realizados estão descritos a seguir:

Teste unitário: Toda nova codificação foi brevemente testada para analisar se apontava algum defeito. Usado a ferramenta Visual Studio Code para mostrar erros de sintaxe no código.

Teste de conteúdo: Todo texto e imagem foram revisados, sendo analisados a coerência e coesão do conteúdo.

Teste de interface: Analisado se o site contém clareza na interatividade com o usuário, observando se houve omissão e/ou ambiguidade dos botões e na apresentação dos cadastros.

Teste de navegabilidade: Analisado se o usuário conseguia acessar cada página e a navegação dele pelo site de forma rápida.

Teste de compatibilidade: Avaliado o sistema em outros navegadores, sendo analisado se as mesmas funções foram executadas em diferentes navegadores.

5. IMPLANTAÇÃO

Link para o github:

- <https://github.com/devDavyRibeiro/ProjetoInterdisciplinar>

Apêndice

```
create database SANEASP
```

```
use SANEASP
```

```
create table Usuarios(
```

```
    id_usuario int not null identity (1,1) primary key,
```

```
    nome_usuario varchar(50) not null,
```

```
    telefone varchar(14),
```

```
    email varchar(40) not null unique,
```

```
    senha varchar(100) not null,
```

```
    cpf char(11) unique,
```

```
    cep char(8),
```

```
    cidade varchar(30),
```

```
    bairro varchar(30),
```

```
    rua varchar(30),
```

```
    numero varchar(30),
```

```
    complemento varchar(30)
```

```
)
```

```
create table Administradores(
```

id_admin int not null identity (1,1) primary key,

id_usuario int not null,

nivel int not null

Constraint fk_usuario_admin foreign key (id_usuario) references
Usuarios(id_usuario)

)

create table Tabela_logs(

id_log int not null identity(1,1) primary key,

operacao varchar(10) not null,

conteudo_log varchar(500) not null,

versao_conteudo varchar(10) not null,

data_log datetime not null default GETDATE(),

id_usuario int not null

Constraint fk_usuario_log foreign key (id_usuario) references
Usuarios(id_usuario)

)

create table Doencas(

id_doenca int not null identity(1,1) primary key,

nome_doenca varchar(50) not null,

descricao_doenca varchar(500) not null,

transmissao varchar(255) not null,

tratamento varchar(255) not null,

data_publicacao_doenca datetime not null default GETDATE(),

id_admin int not null

Constraint fk_admin_doenca foreign key (id_admin) references
Administradores(id_admin)

)

create table Sintomas(

id_sintoma int not null identity(1,1) primary key,

nome_sintoma varchar(50) not null

)

create table Sintomas_doenca(

id_sintoma_doenca int not null identity(1,1) primary key,

id_doenca int not null,

id_sintoma int not null,

Constraint fk_doenca foreign key (id_doenca) references
Doencas(id_doenca),

```

        Constraint    fk_sintoma    foreign    key    (id_sintoma)    references
Sintomas(id_sintoma)

)

```

```

create table Reclamacoes (

```

```

    id_reclamacao int not null identity(1,1) primary key,

```

```

    titulo_reclamacao varchar(50) not null,

```

```

    descricao_reclamacao varchar(500) not null,

```

```

    cep char(8),

```

```

    cidade varchar(30),

```

```

    bairro varchar(30),

```

```

    rua varchar(30),

```

```

    numero varchar(30),

```

```

    complemento varchar(30),

```

```

    status_reclamacao varchar(20) not null,

```

```

    data_reclamacao datetime default GETDATE() not null,

```

```

    pontuacao decimal(5,2) not null,

```

```

    id_usuario int not null,

```

```

        Constraint    fk_usuario_reclamacao    foreign    key    (id_usuario)    references
Usuarios(id_usuario)

)

```

```
create table Comentarios(  
  
    id_comentario int not null identity(1,1) primary key,  
  
    descricao_comentario varchar(500) not null,  
  
    data_comentario datetime default GETDATE() not null,  
  
    id_admin int,  
  
    id_usuario int,  
  
    id_reclamacao int not null,  
  
    Constraint fk_comentario_administrador foreign key (id_admin)  
references Administradores(id_admin),  
  
    Constraint fk_comentario_usuario foreign key (id_usuario) references  
Usuarios(id_usuario),  
  
    Constraint fk_comentario_reclamacao foreign key (id_reclamacao)  
references Reclamacoes(id_reclamacao)  
  
)
```

```
create table Imagens_reclamacao(  
  
    id_imagem_reclamacao int not null identity(1,1) primary key,  
  
    imagem_reclamacao varchar(100) not null,  
  
    id_reclamacao int not null  
  
    Constraint fk_imagem_reclamacao foreign key (id_reclamacao)  
references Reclamacoes(id_reclamacao)
```

)

create table Tags(

id_tag int not null identity(1,1) primary key,

nome_tag varchar(50) not null

)

create table Tags_reclamacao(

id_tag_reclamacao int not null identity(1,1) primary key,

id_tag int not null,

id_reclamacao int not null

Constraint fk_tags_reclamacao_tag foreign key (id_tag) references
Tags(id_tag),

Constraint fk_tags_reclamacao_reclamacao foreign key (id_reclamacao)
references Reclamacoes(id_reclamacao)

)

create table Noticias (

id_noticia int not null identity(1,1) primary key,

titulo_noticia varchar(50) not null,

descricao_noticia varchar(500) not null,

data_publicacao_noticia datetime not null default GETDATE(),

id_admin int not null

Constraint fk_noticia_administrador foreign key (id_admin) references
Administradores(id_admin)

)

create table Imagens_noticia(

id_imagem_noticia int not null identity(1,1) primary key,

imagem_noticia varchar(100) not null,

id_noticia int not null

Constraint fk_imagem_noticia foreign key (id_noticia) references
Noticias(id_noticia)

)

create table Fontes_noticia(

id_fonte_noticia int not null identity(1,1) primary key,

fonte_noticia varchar(255) not null,

id_noticia int not null

Constraint fk_fonte_noticia foreign key (id_noticia) references
Noticias(id_noticia)

)

```
create table Tags_noticia(  
  
    id_tag_noticia int not null identity(1,1) primary key,  
  
    id_tag int not null,  
  
    id_noticia int not null  
  
    Constraint fk_tags_noticia_tag foreign key (id_tag) references  
Tags(id_tag),  
  
    Constraint fk_tags_noticia_noticia foreign key (id_noticia) references  
Noticias(id_noticia)  
  
)
```

```
create table Imagens_doenca(  
  
    id_imagem_doenca int not null identity(1,1) primary key,  
  
    imagem_doenca varchar(100) not null,  
  
    id_doenca int not null  
  
    Constraint fk_imagem_doenca foreign key (id_doenca) references  
Doencas(id_doenca)  
  
)
```

```
create table Fontes_doenca(  
  
    id_fonte_doenca int not null identity(1,1) primary key,
```

fonte_doenca varchar(255) not null,

id_doenca int not null

Constraint fk_fonte_doenca foreign key (id_doenca) references
Doencas(id_doenca)

)