

# Documentação Blog-SQLite 2025

---

30 DE MAIO

**Senai**

**Criado por: Pedro Muniz, Samuel Baek, Beatriz  
Pupo, Yuri Nogueira**



## Sumário

<b>1. Introdução e Descrição do Projeto.....</b>	<b>3</b>
<b>1.1. Sobre o Projeto Analisado:.....</b>	<b>3</b>
<b>1.2. Principais Funcionalidades Identificadas:.....</b>	<b>3</b>
<b>1.3. Funcionalidades não implementadas:.....</b>	<b>3</b>
<b>2. Análise Exploratória e Testes Funcionais.....</b>	<b>3</b>
<b>2.1. Processo de Teste Realizado:.....</b>	<b>3</b>
<b>2.2. Observações Gerais e Usabilidade:.....</b>	<b>3</b>
<b>4. Propostas de Melhoria.....</b>	<b>5</b>
<b>4.1. Correção de Bugs:.....</b>	<b>5</b>
<b>4.2. Melhorias na Segurança:.....</b>	<b>6</b>
<b>4.3. Melhorias na Organização do Código:.....</b>	<b>6</b>
<b>4.4. Melhorias na Documentação:.....</b>	<b>6</b>
<b>4.5. Melhorias no Tratamento de Erros:.....</b>	<b>6</b>
<b>5. Diagramas e Fluxogramas.....</b>	<b>7</b>
<b>5.1. Fluxograma: Criação de Novo Post.....</b>	<b>7</b>
<b>6. Conclusão.....</b>	<b>7</b>
<b>7. Anexos.....</b>	<b>7</b>

# 1. Introdução e Descrição do Projeto

## 1.1. Sobre o Projeto Analisado:

- O projeto analisado é uma aplicação de blog desenvolvida com [Node.js](#), framework Express, Sqlite3 e bodyParser. Seu objetivo principal é postar comentários, fotos e vídeos sobre momentos importantes da sua vida.
- O repositório do projeto está localizado em: [https://github.com/DenadaiSenai/BlogSQLite\\_tutorial\\_reiniciado](https://github.com/DenadaiSenai/BlogSQLite_tutorial_reiniciado).

## 1.2. Principais Funcionalidades Identificadas:

- Cadastro de novos usuários.
- Login de usuários.
- Navegar pelas páginas.
- Acesso a um Dashboard detalhado.
- Logout de usuário

## 1.3. Funcionalidades não implementadas:

- Criação de novos posts (Título e Conteúdo).

- Listagem dos posts existentes.
- Comentários em posts.
- Avaliar posts.
- Seguir novos usuários.
- Personalização de perfil.

## 2. Análise Exploratória e Testes Funcionais

### 2.1. Processo de Teste Realizado:

- O projeto foi clonado e foi realizado a execução local seguindo as instruções do README com as dependências (Nodemon app.js, SQLite3, Express.js) no sistema operacional Linux (Ubuntu 24.04.2 LTS). As funcionalidades foram testadas através do navegador utilizando a rota localhost:9000, tentando diferentes entradas de dados e fluxos de usuário.

### 2.2. Observações Gerais e Usabilidade:

- A interface para fazer cadastro e login é de fácil entendimento assim como para fazer logout e acessar outras áreas, mas apresenta falhas de segurança e problemas em aplicar máscaras de segurança no cadastro e em mostrar dados privados dos usuários.

## 3. Problemas Identificados e Oportunidades de Melhoria

### 3.1. Bugs Funcionais Encontrados:

- **Bug 1:** Botão “Esqueci minha senha”
  - **Descrição:** Na página “/login” não existe botão para recuperar senha em caso de “esqueci minha senha”.
- **Bug 2:** Botão “Voltar”
  - **Descrição:** Na página “/invalid\_login” não existe botão para voltar a página “/home”
- **Bug 3:** Melhorar responsividade página dashboard
  - **Descrição:** Na página “/dashboard” falta responsividade.
- **Bug 4:** Página "SOBRE"
  - **Descrição:** Página “/sobre” não possui conteúdo.

## 3.2. Pontos de Atenção na Segurança (Nível Básico):

- **Bug 1:** Acesso ao “DASHBOARD”
  - o **Descrição:** Ao realizar o login no site o usuário é direcionado automaticamente a página “/dashboard”, fazendo com que todos tenham acesso ao banco de dados
- **Bug 2:** Máscara nos campos de cadastro
  - o **Descrição:** Na página “/cadastro” os dados a serem preenchidos não possuem máscara, fazendo com que o usuário preencha seus dados sem o padrão necessário.

## 3.3. Deficiências na Documentação:

- **Bug 1:** Comentar códigos adicionados
  - o **Descrição:** Adicionar comentários especificando os códigos recém adicionados.
- **Bug 2:** “READ.ME” GitHub
  - o **Descrição:** Adicionar “READ.ME” no GitHub para explicar o projeto, funcionalidade e outras informações.

## 3.4. Tratamento de Erros:

- **Bug 1:** Remover os dados pré-preenchido no campo de cadastro
    - o **Descrição:** Ao acessar a página “/cadastro” os dados já estão pré-preenchidos, fazendo com que o usuário possa acessar um perfil já existente, tendo acesso a dados deste usuário.
  - **Bug 2:** Máscara nos campos de cadastro
    - o **Descrição:** Na página “/cadastro” os dados a serem preenchidos não possuem máscara, fazendo com que o usuário preencha seus dados sem o padrão necessário.
  - **Bug 3:** Página de erros mal definida
    - o **Descrição:** Quando o usuário acessa um endereço inválido ele se depara com uma página de erro na qual não especifica o erro, fazendo com que seja retornado uma página genérica.
-

## 4. Propostas de Melhoria

### 4.1. Correção de Bugs:

- **Erro no Botão “Esqueci minha senha”:**
  - **Proposta:** Adicionar um botão que direciona o usuário para uma página que recupere sua senha através de verificação de e-mail ou sms.
  - **Justificativa:** Isso vai melhorar a experiência do usuário caso esqueça sua senha, e sua confiabilidade com o site.
- **Erro no Botão “Voltar ”:**
  - **Proposta:** Adicionar um botão que volte a página home caso o login seja invalido.
  - **Justificativa:** vai melhorar a interface do site
- **Erro na “responsividade do dashboard”:**
  - **Proposta:** Melhorar a responsividade da página de dashboard, Como ajustar o tamanho da janela de visualização e adicionar um scroll para a janela.
  - **Justificativa:** vai melhorar a experiência do usuário com o site, sendo mais responsivo e visualmente atraente
- **Erro na página “sobre”:**
  - **Proposta:** Adicionar um texto explicativo sobre o intuito e funcionalidades do site
  - **Justificativa:** melhorar a experiência do usuário criando uma área amigável e familiar.

### 4.2. Melhorias na Segurança:

- **Proposta para acesso ao “DASHBOARD”:** "Implementar um acesso ao “DASHBOARD” apenas para usuários administradores, criando um login próprio para este acesso.
  - **Justificativa:** Aumento de segurança, pois o acesso a dados dos usuários fica limitado a perfis de segurança e confiança.
- **Proposta para máscara no campo de cadastros:** Adicionar máscara aos campos de cadastro através de um java script.
  - **Justificativa:** Facilita interface do sistema, tratamento de dados além da padronização dos dados requeridos.

### 4.3. Melhorias na Organização do Código:

- **Proposta para comentar códigos adicionados:** Comentar os códigos recém adicionados.

- **Justificativa:** Formatação e clean code, a fim de facilitar o entendimento do código em futuras aplicações e melhorias.
- **Proposta para máscara no campo de cadastros:** Adicionar máscara aos campos de cadastro através de um java script.
- **Justificativa:** Facilita interface do sistema, tratamento de dados além da padronização dos dados requeridos.

## 4.4. Melhorias na Documentação:

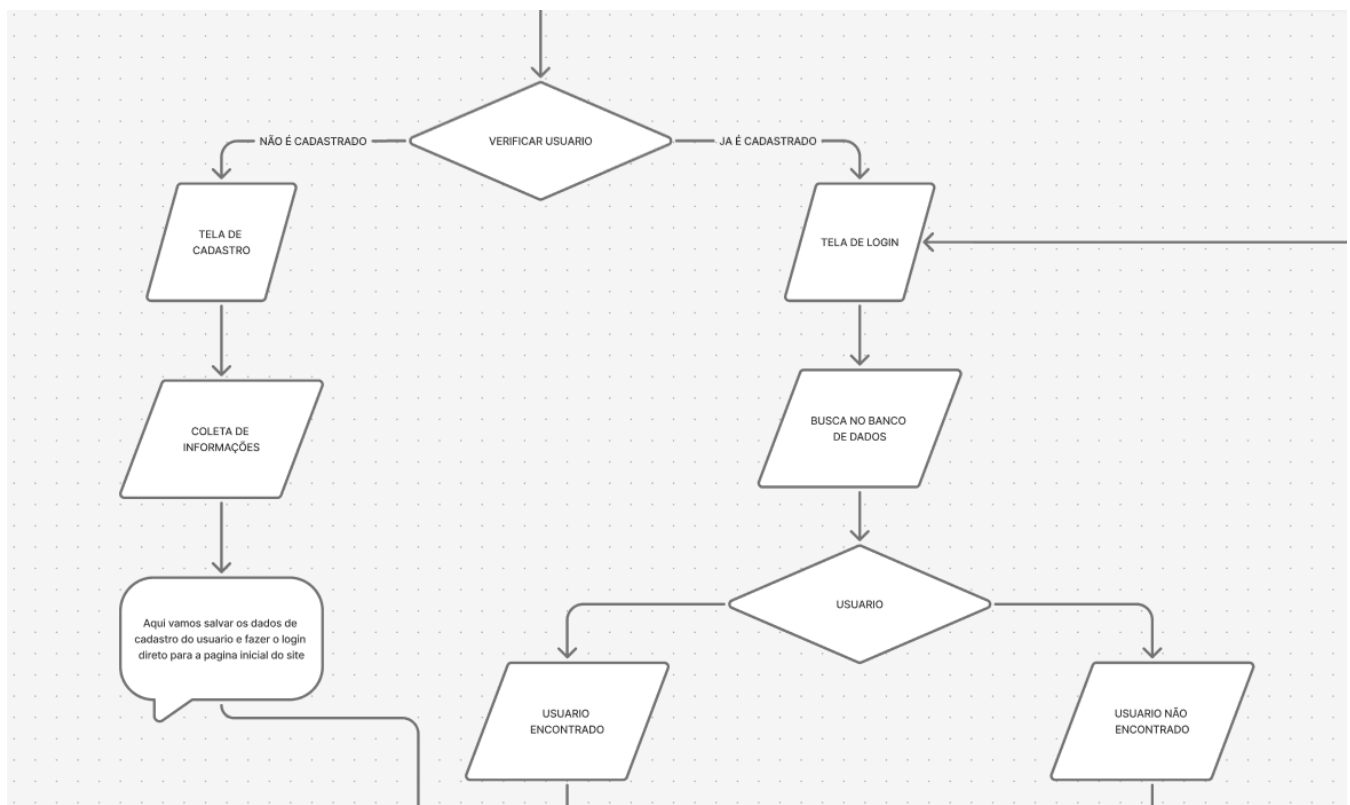
- **Proposta para [README.md](#):** Veja o hyperlink com a proposta de README.md atualizado.

## 4.5. Melhorias no Tratamento de Erros:

- **Proposta:** Criar uma branch no github para melhorar todos os erros já existentes e categorizar as versões futuras (Ex: Versão 1.0, 1.01, 1.02...).
- **Justificativa:** irá dar um grau de profissionalismo ao projeto como também melhorar a visualização de códigos e erros passados.

# 5. Diagramas e Fluxogramas

## 5.1. Fluxograma: Criação de Novo Post



Link para Projeto [Fluxograma](#)

## 6. Conclusão

- Após uma profunda análise do projeto “Blog-SQLite” revelou grandes deficiências de segurança, responsividade, funcionalidade e documentação. Diante disso, é extremamente necessário que haja investimento para a busca de melhorias e otimização, a fim de melhorar a relação do usuário para com a interface do sistema.
-