

**INSTITUTO NACIONAL DE TELECOMUNICAÇÕES**

**INATEL**

**Contatinhos**

Luiz Felipe Pereira Lima, 1206

Rodrigo Rufino Ribeiro, 1098

**Santa Rita do Sapucaí**

**Junho de 2018**



**INSTITUTO NACIONAL DE TELECOMUNICAÇÕES**

**INATEL**

**Contatinhos**

Projeto apresentado ao Instituto Nacional de Telecomunicações como parte das exigências do curso de Engenharia da Computação, como requisito parcial para obtenção da aprovação na disciplina de Integração de Fundamentos (AC318-B).

**Orientador**

Prof. Me. Adauto Mendes B. Júnior

**Santa Rita do Sapucaí**

**Junho de 2018**

1. Introdução

O **Contatinhos** é um cartão de visitas digital com o fim de facilitar o compartilhamento do contato pessoal de um usuário. A plataforma consiste em apresentar uma página web pessoal, possibilitando ao usuário enviá-la facilmente a outras pessoas para as quais deseja divulgar seu contato. Na página, há um link que possibilita a adição automática do contato na agenda do smartphone.

Esse projeto tem como relevância a facilidade de compartilhar contatos com outras pessoas, já que o modo atual de compartilhamento é trabalhoso e nada prático.

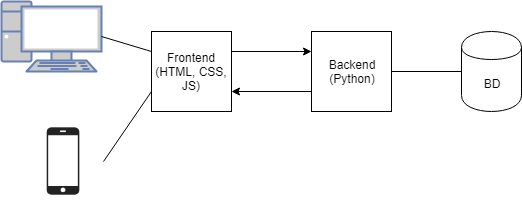
1. JUSTIFICATIVA

Esse sistema fornece uma forma prática e rápida de salvar contatos na agenda do celular. No modo tradicional, o fluxo de adição de um contato à agenda do celular é mais ou menos o seguinte: a pessoa pede o contato para outra; ela abre a agenda do celular; a outra pessoa recita o número (muitas vezes tendo que repetir); ela pede seu nome de contato; a outra pessoa fala seu nome e sobrenome; ela pede seu email... Com o **Contatinhos** todo esse trabalho pode ser resumido em abrir um link no navegador. Esse link é intuitivo pois, como grandes redes sociais, é composto do nome da plataforma seguido do nome de usuário, o que facilita mais ainda o compartilhamento caso tenha que falar o endereço na plataforma. Depois disso, todo o processo é feito automaticamente em questões de segundos. Todos os dados que foi configurado no perfil de usuário será salvo na agenda do celular de outra pessoa.

1. ARQUITETURA DO SISTEMA

**3.1. Esquema Macro do Sistema**

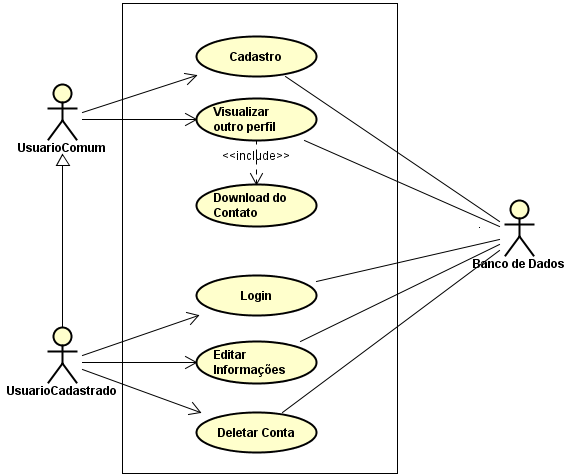
O sistema pode ser definido pelo esquema macro a seguir.



O sqlite é o banco de dados onde as informações sobre os usuários são armazenadas. Elas são acessadas através do Backend, que é responsável por gerir toda a estrutura do sistema (como os dados que serão enviados aos usuários, o que será escrito no banco de dados, como as páginas e seu fluxo são, entre outros). O Backend é a parte do sistema que faz, de fato, o funcionamento ser possível. O Frontend é o que possibilita a interação do usuário com o sistema. É responsável por apresentar a Interface de Usuário e enviar ao Backend todas as interações que o usuário realiza. Ele pode acessar o site através de uma plataforma mobile ou desktop.

**3.2. Diagrama de Casos de Uso**

O modo que o sistema pode ser utilizado é detalhado no Diagrama de Casos de Uso a seguir.



**3.2.1. Descrição dos Atores**

**A1 – UsuarioComum**

O usuário comum é alguém que esteja acessando ao site e que não possui um cadastro.

**A2 – UsuarioRegistrado**

O usuário registrado é alguém que possua um cadastro no site.

**A3 – Banco de Dados**

O Banco de Dados é onde os dados de todos os usuários do sistema é armazenado.

**3.2.2. Descrição dos Casos de Uso**

**Ca1 – Cadastro**

Esse caso de uso possibilita que o usuário se cadastre no sistema, informando dados como telefone, email, username e senha.

**Ca2 – Visualizar outro perfil**

Esse caso de uso possibilita que o usuário visualize perfis de outras pessoas através do link direto do perfil ou através de uma busca.

**Ca3 – Download do Contato**

Esse caso de uso possibilita que o usuário baixe as informações de outra pessoa e as salvem na agenda do seu smartphone.

**Ca4 – Login**

Esse caso de uso possibilita que o usuário faça login no sistema.

**Ca5 – Editar Informações**

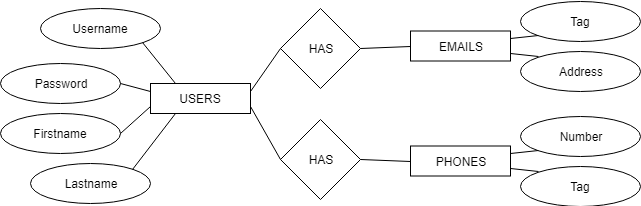
Esse caso de uso possibilita que o usuário edite suas informações.

**Ca6 – Deletar Conta**

Esse caso de uso possibilita que o usuário apague sua conta.

**3.3. Modelo Entidade-Relacionamento**

O seguinte diagrama do Modelo Entidade-Relacionamento mostra como é estruturado o banco de dados do sistema.



Um usuário precisa ter um username e um password. Isso possibilitará que ele realize o login no sistema e crie sua conta. A partir daí, cada usuário pode ter diversos dados, como nome (composto de nome e sobrenome), emails (composto de uma tag e o email) e telefones (composto de uma tag e o email), todos estes serão adicionados na agenda de contatos automaticamente quando alguém entrar no perfil do usuário e escolher salvar o contato.

1. Metodologia

O sistema consiste de uma aplicação desenvolvida em Flask (microframework web em Python) que recebe os requests e envia respostas ao cliente. Com estrutura monolítica, o servidor entrega tanto as páginas que são renderizadas no cliente (frontend), quanto gerenciar os dados recebidos e armazená-los em banco (backend). Para desenvolvimento local é utilizado um banco de dados SQLite3, enquanto a versão em deploy na plataforma Heroku utiliza PostgreSQL, que é provisionado gratuitamente.

A aplicação também implementa o padrão de Progressive Web App, que permite a instalação em dispositivos diretamente do navegador e algumas funcionalidades offline. Isto se dá pela adição de um manifesto de aplicação (JSON) e um service worker (Javascript) que armazena dados selecionados em cache.

1. Resultados

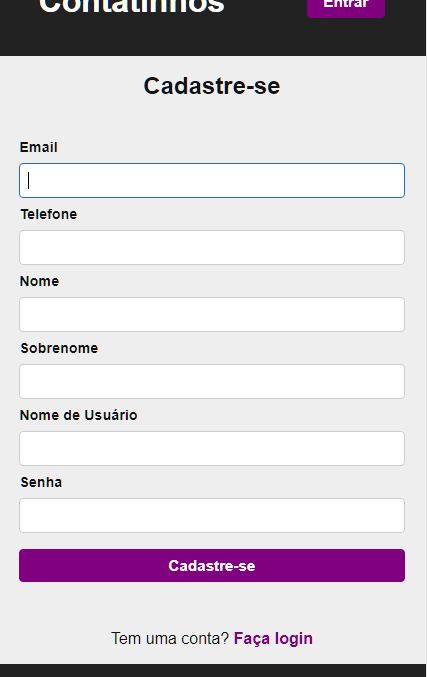
**5.1. Tela de login**

Nessa tela, o usuário pode fazer o seu cadastro ou o login. Se relaciona aos casos de uso **Ca1** e **Ca4**.



**5.2. Tela de Cadastro**

Essa tela permite que o usuário realize o cadastro no sistema. O botão no cabeçalho também permite que ele realize o login. Se relaciona aos casos de uso **Ca1** e **Ca4**.



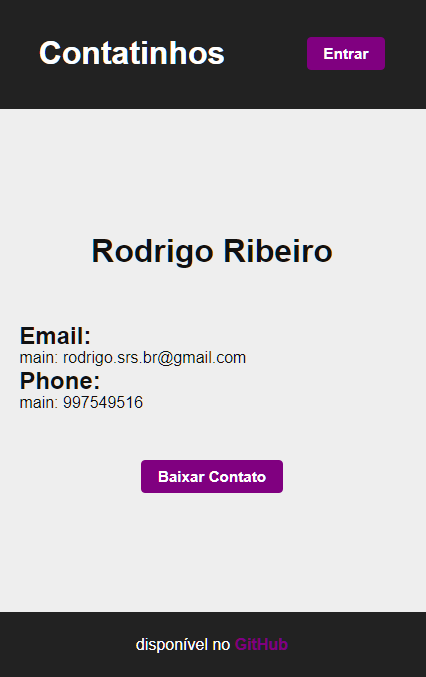
**5.3. Perfil (após login)**

Após realizar o login, o usuário poderá visualizar suas informações nessa tela, assim como sair do sistema, editar suas informações, e baixar seu próprio contato. Se relaciona aos casos de uso **Ca2**, **Ca3** e **Ca5**.



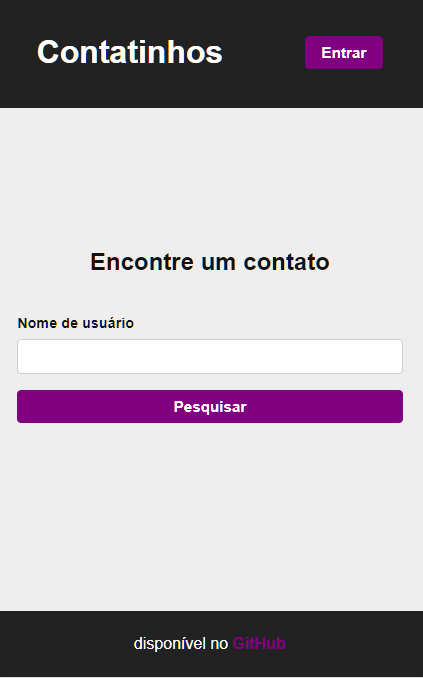
**5.4. Perfil (visitar perfil de outro usuário)**

Quando entra no link do perfil de alguma pessoa, o usuário pode visualizar as informações e/ou baixar o contato dessa pessoa. Se relaciona aos casos de uso **Ca2** e **Ca3**.



**5.5. Buscar por Usuários**

O usuário pode realizar a pesquisa por outros usuários. Se relaciona ao caso de uso **Ca2**.



1. PROPOSTA PARA TRABALHOS FUTUROS

O projeto ainda poderia conter algumas funções como:

* **Adição de Fotos:** Permitir a adição de foto de perfil no cartão de visitas.
* **Personalização do Cartão de visitas:** Permitir que o usuário personalize sua página com imagem de background, cores diferentes, fontes diferentes, entre outros.(background, cores, imagens, fontes)
* **Integração com redes sociais:** Permitir a integração com outras redes sociais para que elas também apareçam no perfil do usuário

1. Referências bibliográficas

RONACHER, Armin. **Flask - web development, one drop at a time.**2010. Disponível em: <http://flask.pocoo.org/>. Acesso em: 01 jan. 2010.