

Apresentação da documentação do sistema Solus

Angelo Silva

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo Câmpus Boituva
Curso de Análise e Desenvolvimento de sistemas

27 de março de 2018, v-0.0.1

Sumário

- 1 Construção
- 2 Introdução
- 3 Descrição geral do sistema
- 4 Requisitos do sistema
- 5 Tecnologias empregadas
- 6 Exemplo de filtros

Construção

A documentação do projeto foi construída utilizando `lateX` e o framework `abntex`.

O projeto pode ser encontrado em: **github.com/r6d6/solus**

Introdução

Na sociedade contemporânea, diversas preocupações quanto a captação de energia surgiram. Uma dessas preocupações é cada vez mais, buscar fontes renováveis de energia.

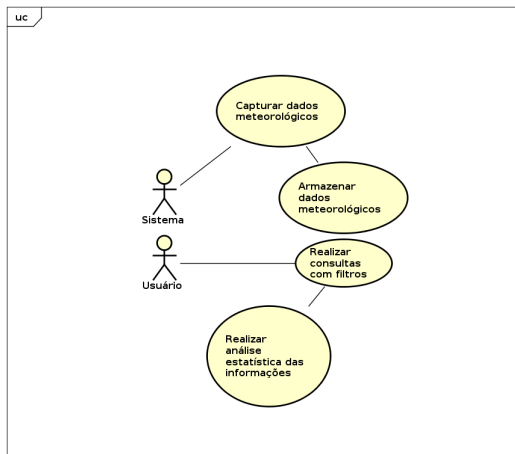
Descrição geral do sistema

O projeto visa, através da análise estatística de dados meteorológicos, auxiliar o estudo de viabilidade acerca da instalação de painéis fotovoltaicos.

Para isso, serão coletados dados através de sensores conectados a um microcontrolador arduino. Inicialmente, prevemos captar informações de umidade do ar, temperatura e incidência de radiação solar.

Requisitos do sistema

Figura: Diagrama de caso de uso



Requisitos não funcionais

- Segurança
- Disponibilidade
- Performance ...

Tecnologias empregadas

- Microcontrolador arduino e suas libraries para requisição HTTP
- GIT como sistema de versionamento para versionamento
- PHP para criação da API
- Lumen como framework para construção da API
- Percona Server como SGBD (MySQL)
- Docker para gerenciamento dos containers da aplicação ...

Regras de negócio

A maior parte das regras de negócio de sistema fica centralizada nos filtros, eles são quem validam os dados do sistema, definindo regras para a captura de dados. A regra de negócio de filtragem diz que os dados de umidade e temperatura não podem se diferenciar por 50% ou mais da média dos dados captados na ultima hora.

```
$data->addFilter(function($data) use ($highLimit) {  
    return $data < $highLimit * 1.30;  
})->addFilter(function($data) use ($lowLimit) {  
    return $data > $lowLimit * 1.30;  
});
```