|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Arquitetos:  Mena Simão  Murilo Machado  Samuel Armbrust  Autores:  Mena Simão  Murilo Machado  Samuel Armbrust | **Descrição Funcional**  **do Produto**   |  | | --- | | **Estação Meteorológica** | | | | |
|  | Versão | Estado | Data | Revisor/Aprovado por |
|  | 0.1 | Em andamento | 23/09/18 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  | | | |

**Conteúdos**

[**Informações Gerais**](#_30j0zll) **3**

[**Referencias**](#_3znysh7) **3**

[**Design**](#_tyjcwt) **3**

[**Objetivos do Design**](#_3dy6vkm) **3**

[**Contexto**](#_1t3h5sf) **3**

[**Blocos principais do projeto**](#_4d34og8) **3**

[**Requerimentos do projeto**](#_17dp8vu) **3**

[**Desafios do design**](#_gtrnab1z6axe) **3**

[**Interface do usuário**](#_3rdcrjn) **3**

[**Frameworks / Componentes utilizados**](#_26in1rg) **3**

[**Documentação dos Detalhes do Design**](#_qsh70q) **4**

[**Banco de Dados**](#_3as4poj) **4**

[**Software**](#_6giuqwbk8s90) **4**

[**Hardware**](#_6qjc01avx7ns) **4**

[**Padrões de Design Implementados**](#_49x2ik5) **4**

[**Apêndice**](#_3o7alnk) **4**

[**Glossário**](#_23ckvvd) **4**

[**Análise de Erros**](#_1hmsyys) **4**

[**Ferramentas utilizadas**](#_b3yqsnaf8q1) **4**

# Informações Gerais

## Referencias

|  |  |  |
| --- | --- | --- |
| **Documento** | **Data** | **Link** |
| DHT11 | 20/09/2018 | http://www.ocfreaks.com/basics-interfacing-dht11-dht22-humidity-temperature-sensor-mcu/ |
| BMP180 | 20/09/2018 | https://learn.sparkfun.com/tutorials/bmp180-barometric-pressure-sensor-hookup- |

# Design

## Objetivos do Design

A estação meteorológica que deverá fornecer os dados de:

* temperatura - LM35 (DHT11)
* umidade relativa - DHT11
* radiação solar - LDR
* pressão atmosférica - Sensor Pressão Atmosférica Barômetro Bmp180 Arduino / Pic
* horas de radiação solar por dia

A estação deverá funcionar continuamente, sua instalação não ficará próxima a alguma fonte de alimentação.

* Bateria + Placa Solar

Deverá ser possível acessar as últimas N medições de alguma forma fácil e amigável ao usuário leigo.

* Módulo bluetooth para conectar

Qual a melhor forma de amostrar?

* Ler sobre estações meteorológicas e frequência de medição

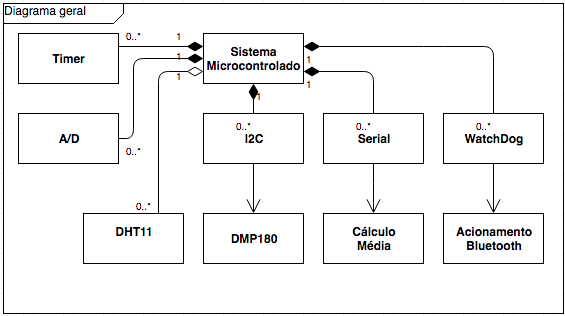
## Escopo

* + - * Produto deve ser o mais próxima da produção
      * Sistema de análise meteorológica
      * Medir temperatura
      * Medir luminosidade
      * Medir umidade
      * Medir pressão atmosférica
      * Medir horas de radiação por dia
      * Funcionará com bateria
      * Terá sistema de carregamento a partir de placa fotovoltaica
      * Interface do usuário
* Exibir valores de medição das últimas 24 horas
* 4 Medições por hora -> gera média
* Utilizará bluetooth para comunicar com o sistema e o celular

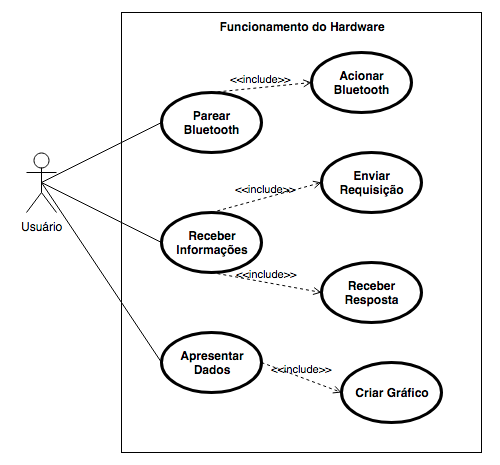
## Contexto

## Principais blocos do projeto

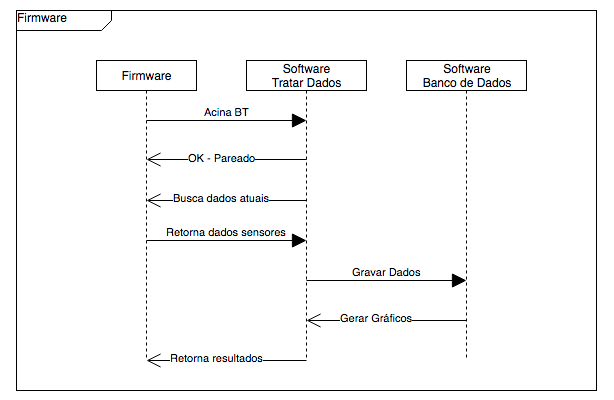
* + - Hardware
      * Diagrama de blocos do Firmware



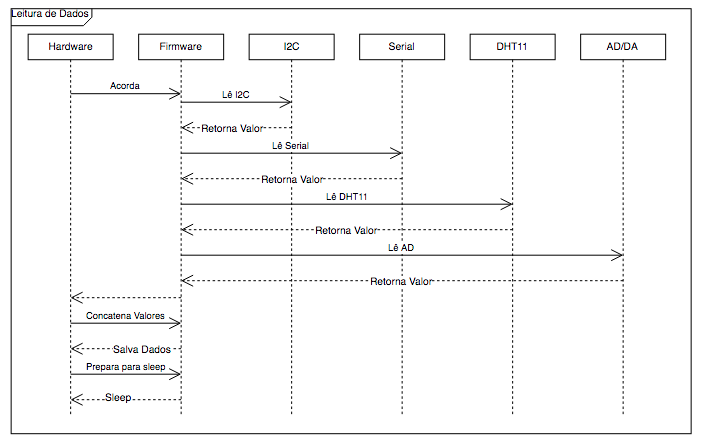
* + - * Diagrama Caso de Uso do Firmware



* + - * Diagrama de Sequência do Firmware



* + - * Diagrama de Sequência da Leitura de dados do Firmware



## Requerimentos do projeto

Requerimentos funcionais:

* Sistema deve ter aplicativo
* Sistema deve exibir valores de medição
* Sistema deve fazer média de leituras em cada hora
* Registrar tempo de luminosidade
* Deve ter sinalização em função da altura

## Desafios do design

Requerimentos não funcionais:

* Sistema deve ter bateria
* Sistema deve operar em baixo consumo
* Sistema terá acionamento para o bluetooth - diminuir consumo
* Sistema deve ter gráfico com 24 pontos
* Será um aplicativo android

## Interface do usuário

* Como conectar bluetooth com celular
* Como criar app que use bluetooth

## Frameworks / Componentes utilizados

|  |  |
| --- | --- |
| **Nome** | **Descrição** |
|  |  |
|  |  |

# Documentação dos Detalhes do Design

## Banco de Dados

## Particionamento Hardware / Software

Idealmente, o projeto será dividido em duas threads, começando pela implementação do hardware e depois pela implementação do software. Com isso, uma parte do time de desenvolvimento dividirá as atividades no M4 e outra na aplicação Android.

## Protocolo de Comunicação

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bit 1 | Bit 2 | Bit 3 | Bit 4 | Bit 5 | Bit 6 | Bit 7 | Bit 8 |
| Numeral | Decimal | Numeral | Decimal | Numeral | Decimal | Numeral | Numeral |
| RH | RH | Temp | Temp | Bmp | Bmp | LDR | LDR |

## Software

* Android
* Material Design
* Atollic
* STM32CubeMX

## Implementação do Software

A fim de projetar a lista de tarefas a serem executadas, dividiremos a implementação do software nas seguintes tarefas:

* Criar projeto básico Android
* Implementar a comunicação bluetooth
* Executar teste para parear com dispositivo de hardware
* Implementar envio de dados bluetooth
* Implementar protocolo de envio determinado
* Teste de execução do protocolo
* Buscar modelos de interface gráfica
* Implementar interface gráfica
* Verificar necessidade de navegação
  + Se houver navegação, implementá-la
* Teste fim-a-fim

## Hardware

* M4 (ARM)
* LDR (analog)
* DHT11 (serial) [Documentação Protocolo](http://www.ocfreaks.com/basics-interfacing-dht11-dht22-humidity-temperature-sensor-mcu/)
* Bluetooth HC05 (serial)
* Bmp180 (i2c) [Documentação Protolo](https://learn.sparkfun.com/tutorials/bmp180-barometric-pressure-sensor-hookup-)
* Bateria 9V (com regulador)
* Bateria de BACKUP

## Implementação do Hardware

Da mesma forma, para o hardware, as tarefas a seguir buscam a completa implementação do escopo determinado:

* Criar projeto no STM32CubeMX
  + Configurar ADC, I2C e Serial
* Gerar código para o Atollic
* Criar primeira configuração na protoboard com o M4
* Implementar comunicação com o LDR através da ADC
* Teste de comunicação com LDR
* Implementar comunicação serial para o DHT11
* Teste de comunicação com DHT11
* Implementar comunicação Bluetooth
* Teste da comunicação Bluetooth
* Teste de integração com software
* Implementar comunicação I2C com BMP180
* Teste de comunicação com BMP180
* Projetar circuito da bateria
* Testar circuito da bateria de forma independente
* Testar no M4 com regulador de tensão

## Padrões de Design Implementados

# Apêndice

## Glossário

|  |  |  |
| --- | --- | --- |
| **Termo** | **Abreviação** | **Definição** |
|  |  |  |
|  |  |  |

## Análise de Erros

### Ferramentas utilizadas