



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**Departamento de Engenharia de Teleinformática**

# **Smart Aquário**

**Equipe:** Daniel Araújo Chaves Souza  
Everton Freitas da Silva  
Guilherme Alves de Araújo  
Magno Felipe Távora da Silva

Fortaleza, 22 de Novembro de 2019

# Índice

<b>1 Glossário</b>	<b>3</b>
<b>2 Histórico de Revisões</b>	<b>4</b>
Fase 1:	5
Fase 2:	5
Fase 3:	6
Fase 4:	6
<b>3 Materiais</b>	<b>6</b>
<b>4 Descrição</b>	<b>7</b>
<b>5 Contextualização do Projeto</b>	<b>8</b>
5.1 Objetivo	8
5.2 Público alvo	8
<b>6 Arquitetura</b>	<b>8</b>
<b>7 Padrões de Projeto</b>	<b>9</b>
<b>8 Conclusão</b>	<b>10</b>
<b>9 Perspectivas futuras</b>	<b>10</b>
<b>10 Referências/bibliografia</b>	<b>11</b>

## 1 Glossário

Siglas	Definição
IoT	Internet of Things
AWS	Amazon Web Services
MQTT	Message Queuing Telemetry Transport
VM	Virtual Machine
EC2	Elastic Compute Cloud
SO	System Operacional
LAN	Local Area Network
IPv4	Protocol of Internet version 4

## 2 Histórico de Revisões

Data	Versão	Descrição	Responsável
25/09/19	1	Teste da RaspBerry Pi 3 e Sensor de Temperatura	Magno Felipe, Daniel Araújo, Everton Freitas
25/09/19	1	Teste Servo Motor, Sensor de Nível e Fita de led	Magno Felipe, Daniel Araújo
25/09/19	1	Análise Broker AWS	Guilherme Alves
14/10/19	1	Inclusão dos diagramas de sequência	Daniel Araújo
15/10/19	1	Inclusão das interfaces do sistema	Everton Freitas
15/10/19	1	Inclusão das interfaces do sistema	Guilherme Alves
15/10/19	1	Inclusão do material e do padrão do projeto	Magno Felipe
17/10/19	1	Criação da Identidade visual do projeto	Daniel Araújo
18/10/19	1	Desenvolvimento Firmware	Magno Felipe
18/10/19	1	Desenvolvimento modelo 3D alimentador	Daniel Araújo

25/10/19	1	Desenvolvimento protótipo do aplicativo	Everton Freitas
25/10/19	1	Montagem da parte eletrônica	Daniel Araújo e Magno Felipe
28/10/19	1	Testes do Firmware desenvolvido e comunicação	Guilherme Alves, Daniel Araújo, Magno Felipe
28/10/19	1	Montagem e testes de hardware	Daniel Araújo, Guilherme Alves
29/10/19	2	Desenvolvimento Firmware	Magno Felipe
30/10/19	2	Testes do Firmware desenvolvido e comunicação	Guilherme Alves, Daniel Araújo, Magno Felipe
01/11/19	1	Desenvolvimento aplicativo	Everton Freitas, Magno Felipe e Guilherme Alves
05/11/19	3	Teste do Firmware e Hardware	Daniel Araújo e Magno Felipe
09/11/19	1	Desenvolvimento de suporte para o hardware no aquário	Daniel Araújo
09/11/19	2	Desenvolvimento e testes do aplicativo	Everton Freitas e Magno Felipe
10/11/19	1	Teste de comunicação entre sensores, atuadores e aplicativo	Daniel Araújo, Everton Freitas, Guilherme Alves e Magno Felipe
14/11/19	3	Desenvolvimento e ajustes finais do Firmware definitivo	Daniel Araújo
20/11/19	1	Montagem dos componentes de hardware no suporte do aquário	Daniel Araújo, Guilherme Alves
20/11/19	2	Testes dos componentes eletrônicos e da comunicação com o broker	Daniel Araújo, Magno Felipe, Everton Freitas e Guilherme Alves
20/11/19	3	Finalização do desenvolvimento do aplicativo	Daniel Araújo, Magno Felipe e Everton Freitas
21/11/19	1	Documentação do projeto	Guilherme Alves

Dentro do histórico do projeto, pode se notar que ao longo do tempo o projeto foi dividido em fases, para um melhor controle de seu desenvolvimento..

## ❑ Fase 1:

Na primeira fase do projeto, Escolhemos o tema do projeto, e sua ideia principal em sequência desenvolvemos todo um brainstorming de idéias, enumerando os pontos que poderiam ser trazer problemas e outros que já poderiam ser solucionados. Nessa fase foi onde definimos nossa arquitetura de projeto, os sensores e atuadores que teriam dentro da nossa aplicação, como seria desenvolvido nosso aplicativo numa chamada versão inicial, além da escolha do broker para estabelecer a comunicação, com isso foi elaborado como seria o plano de ação e como seriam divididas as tarefas para mantermos tudo dentro do prazo, além de criamos um repositório git para controle de versionamento do projeto.

## ❑ Fase 2:

Na segunda fase, foi onde nosso projeto começou a sair do papel, fizemos o esquemático elétrico do projeto e uma simulação para ver se tudo estava como planejado, em seguida começamos a fazer as montagens das partes eletrônicas, aliado a isso já fomos fazendo os diversos testes necessários para corrigir bugs que surgiam. Foi nessa fase também que fizemos uma pesquisa mais aprofundada sobre qual broker e como utilizá-lo decidindo por usar serviços da AWS juntamente com o broker Mosquitto. Ocorreu também a inicialização da criação do aplicativo, como definição das telas iniciais e compra de todos os materiais necessários para elaboração do nosso projeto, com isso deu se início também o desenvolvimento do firmware de nossa aplicação, aliado a isso a criação de uma identidade visual para um projeto faz toda diferença como logomarca, cores dentre outras características.

## ❑ Fase 3:

Na fase 3, já com o aquário em mãos, fruto de uma parceria, podemos esboçar melhor nosso protótipo do projeto, além de modelar nosso alimentador na impressora 3D. Fizemos novos testes nos componentes eletrônicos, além de começar a testar esses componentes com o firmware desenvolvido, num teste de hardware completo, simulando a comunicação de um broker local com os atuadores e sensores. Deu início a codificação do aplicativo utilizando o framework React-Native, além de assinarmos um pacote free do serviço da Amazon. Nessa fase ocorreram diversos testes, a fim de tornar o projeto mais fiel possível à realidade, já com o desenvolvimento do aplicativo avançado podemos fazer testes que integravam tanto o hardware como nosso aplicativo, estabelecendo sua comunicação com o broker e vendo se obtinham respostas com nossos sensores e atuadores, fazendo testes de atuação e captura de dados a fim de finalizar o projeto.

## ❑ Fase 4:

Na última fase do projeto, se deu os ajustes finais, foi desenvolvido o protótipo de suporte para o hardware no aquário na qual ele ficaria acima com todos os componentes conectados a nossa Raspberry juntamente com a nossa fonte de alimentação. Nessa última fase também foi feito a montagem final da parte eletrônica, já no local em que ela vai ficar e atuar, o aplicativo foi sendo finalizado juntamente com toda a documentação sobre o projeto. Ademais, serão feitos testes finais tanto de software como de hardware para conferir se o projeto está de acordo com o estipulado, todos os códigos finalizados e documentos foram colocados no git para acesso de todos.

## 3 Materiais

- Sensores:
  - Sensor de umidade ds18b20 (à prova d'água).
  - Sensor de nível do tipo cachimbo.
- Atuadores:
  - Módulo relé.
  - Fita de Led.
  - Motor servo.
- Controladores:
  - Raspberry Pi3.
- Plataformas:
  - Raspbian (SO).
  - node-Red.
  - AWS IOT.
  - AWS Cloud
  - AWS EC2
  - Mosquitto Broker Websocket
- Frameworks:
  - React-Native.
- Outros componentes:
  - jumpers.
  - resistores.
  - protoboard.
  - aquário.
  - transistores.

## **4 Descrição**

Com o advento de tecnologias como o WiFi e a popularização de vários sensores, bem como o acesso à pequenos computadores (Microcontroladores embarcados) surgiu a oportunidade de monitorar praticamente de tudo, inclusive nossos pets. Pensando nessa ideia, surgiu o projeto de um Smart Aquário , no qual este será equipado com vários sensores e atuadores, como sensor de temperatura ds18b20, sensor de nível de água, assim como atuadores como o relé que aciona a bomba de água, o servo motor, que controla o alimentador e o controle de luminosidade controlado pelo usuário no aplicativo, tudo isso utilizando uma RaspBerry PI 3 dotado de um processador Quad Core 1.2GHz Broadcom BCM2837. Utilizamos os serviços da Amazon EC2 para criação da Máquina Virtual com o sistema Linux Ubuntu 16.04 LTS e o Mosquitto Broker Websocket para comunicação entre o aplicativo e o controlador do aquário (Raspberry Pi), com a raspberry e nosso aplicativo. O Smart Aquário será capaz de automatizar grande parte dos cuidados que devem ser tomados para manter o bem-estar dos peixes tudo isso ao controle do usuário pelo seu próprio smartphone.

## **5 Contextualização do Projeto**

### **5.1 Objetivo**

O Smart Aquário surgiu com uma ideia de proporcionar aos amantes de aquários uma experiência inovadora de interação com seu aquário, proporcionando um maior controle sobre o habitat de seus peixes através de seu smartphone, onde ele estiver. O usuário do Smart Aquário será capaz de:

- Controlar a alimentação de seus peixes através de um alimentador automático que pode ser acionado pelo aplicativo, garantindo a alimentação adequada de seus pets;
- Controlar a iluminação do aquário;
- Monitorar a temperatura e programar alertas;
- Monitorar o seu nível de água.

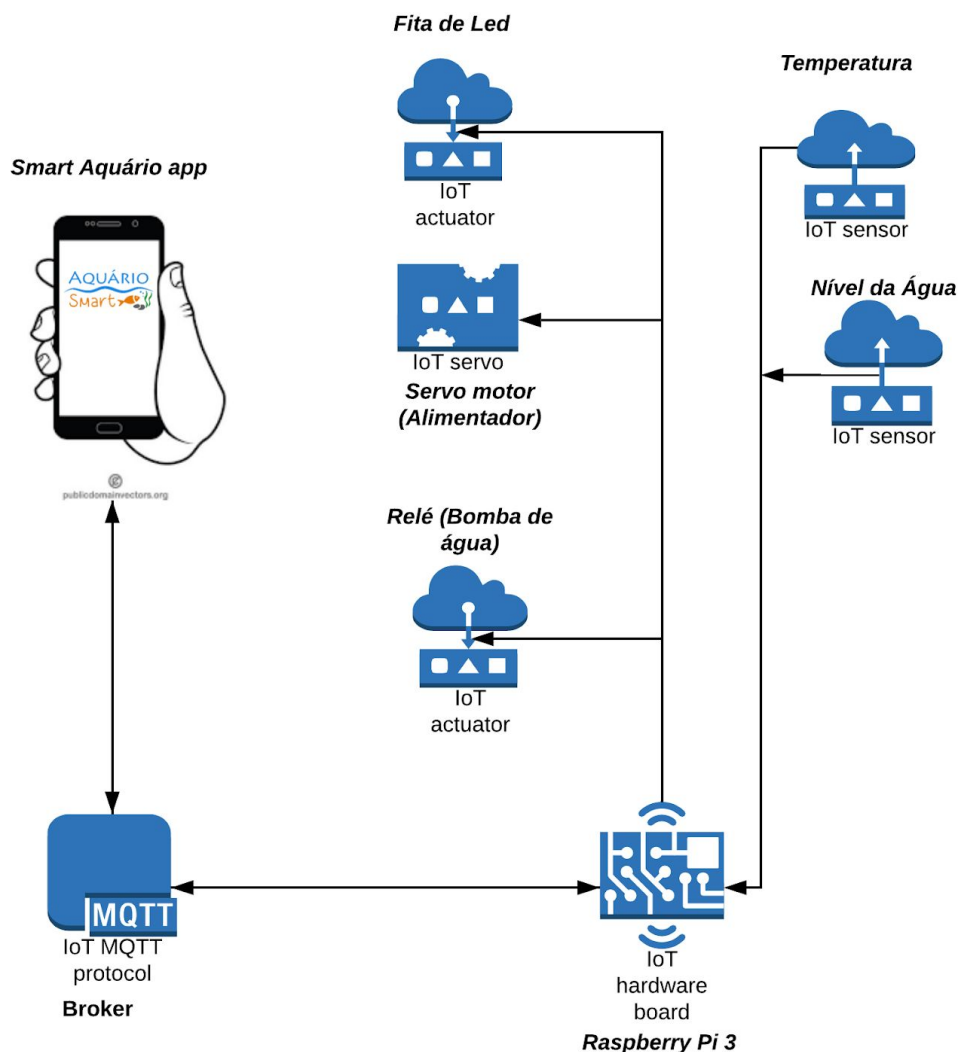
O diferencial de nosso projeto está em tratar-se de um aquário smart que difere da maioria existem pelo leque de funcionalidades que proporciona ao aquarista um ambiente totalmente customizável e controlado.

### **5.2 Público alvo**

Proprietários de aquários de água doce ou salgada, de médio à grande porte.

## 6 Arquitetura

O sistema será composto por uma placa de controle (RaspBerry PI 3), acoplada ao aquário, que coleta os dados dos sensores, aciona os atuadores, quando necessário e faz a comunicação com o broker na nuvem. Esse broker é o Mosquitto Broker Websocket que utiliza o protocolo MQTT que estabelecer a comunicação entre o aplicativo e o controlador do aquário, a Raspberry Pi e nosso aplicativo. O broker mosquitto fica em uma máquina virtual criada no serviço EC2, da Amazon AWS onde utilizamos o Linux Ubuntu 16.04 LTS e dentro desse sistema é onde está no broker.





## **7 Padrões de Projeto**

O Projeto é dividido em 3 camadas, sendo elas a camada de percepção, a camada de rede e a camada aplicação.

A camada de percepção, é a camada que está diretamente em contato com o aquário, é nessa camada que trabalhamos com os componentes físicos do projeto, e é nela que realizamos o sensoriamento do projeto, para saber quais atuadores devem ser ativados para o melhor funcionamento do projeto, nela é onde temos o objetos inteligentes em si, como o nome já diz é nessa camada onde pode se perceber a atuação destes objetos. Já na camada de rede, temos as nossas tecnologias de comunicação, que são para o endereçamento de rede o IPv4, para a comunicação o protocolo MQTT, para a transmissão utilizamos uma rede Wi-fi ou LAN, internet cabeada. Além disso utilizamos da plataforma AWS para emular uma VM e nela hospedar nosso broker Mosquitto, para prover as nossas configurações de Broker, e Cloud. Por último na camada de Aplicação temos os serviços para o cliente, que no nosso caso é o aplicativo mobile, que fornece um gerenciamento de cada aquário do usuário, uma visualização da temperatura do aquário, uma programação de alimentação do peixe (ou ativação automática da mesma), um controle sobre a bomba e sobre o nível de água, é nessa camada que o usuário final interage, na qual para ele as outras duas camadas ficam ocultas.

## **8 Conclusão**

Toda a parte de hardware e firmware foi finalizada, assim como o desenvolvimento do aplicativo. O projeto foi sonhado e concretizado, ocorreram muitos percalços no caminho, dificuldades foram superadas e barreiras quebradas. Para os amantes de aquários, o Smart Aquário poderá revolucionar esse meio. Os desafios propostos foram entregues e uma nova aplicação pode estar surgindo em um mercado emergente.

## 9 Perspectivas futuras

Um mercado crescente como o de aquário, nos faz pensar em perspectivas futuras, a fim de transformar o Smart Aquário em algo ainda mais complexo ao mesmo tempo que simples para quem utiliza. Melhorias podem e devem ser feitas como futuros passos, como por exemplo um novo alimentador com controle de diferentes tipos de rações para diferentes tipos de peixes, além de substituições de sensores a atuadores, por outros melhores e eficientes, abre-se um leque de possibilidades para levarmos o projeto a frente em busca de uma nova e inteligente solução para o mercado.

## 10 Bibliografia

Meu Aquário Conectado. **Minha Casa Conectada**. 2017. Disponível em: <<http://minhacasaconectada.blogspot.com/2017/01/meu-aquario-conectado.html>>. Acesso em: 04 set.2019.

Amazon. **Amazon EC2**. 2019. Disponível em: <<https://aws.amazon.com/pt/>>. Acesso em: 18 out.2019

Raspberry. **Raspberry Pi 4** 2019. Disponível em: <<https://www.raspberrypi.org/>> Acesso em: 18 out.2019

BAUERMEISTER, Giovanni. **Use o Botão IoT e Raspberry Pi para monitorar seu aquário**. 2018. Disponível em: <<https://www.filipeflop.com/blog/monitore-aquario-botao-iot-e-raspberry-pi/>>. Acesso em: 22 set.2019.

CloudMQTT. **Hosted message broker for the**

**Internet of Things.** 2017. Disponível em:  
<<https://www.cloudmqtt.com/docs/index.html>>. Acesso em: 22 set.2019.

React Native. **A framework for building native apps using React.**  
2019. Disponível em:  
<<https://facebook.github.io/react-native/docs/getting-started>>. Acesso em: 18 out.2019.

MQQT. **MQTT pelo protocolo WebSocket.** 2019. Disponível em:  
<[https://docs.aws.amazon.com/pt\\_br/iot/latest/developerguide/mqtt-ws.html](https://docs.aws.amazon.com/pt_br/iot/latest/developerguide/mqtt-ws.html)>.  
Acesso em 18 out.2019.

Node-RED. **Getting Started.** 2018. Disponível em:  
<<https://nodered.org/docs/getting-started/>>. Acesso em: 25 out.2019.