**Universidade de Brasília - UnB**

**Especificação de Arquitetura de Software**

**Orientadores:**

**Alex Reis, Guilherme Bestar, Rhander Viana,**

**Ricardo Chaim, Sébastien Rondineau**

**Brasília, DF**

**Abril de 2018**

# Integrantes de Software

Adailson Santos; Lucas Amoêdo; Leonardo Sagmeister; Marcelo Martins.

# Referencial Teórico

# 1.1. Conceito Arquitetura de Microsserviços

É um modelo arquitetural que tem como foco principal o baixo acoplamento dos serviços disponíveis e é altamente escalável. Ela foi criada para desacoplar as funcionalidades da arquitetura monolítica, sendo esta uma arquitetura tradicional que tem um forte acoplamento entre as classes, o que dificulta a manutenção, aumenta os custos e o tempo do projeto. Com a utilização da arquitetura de microsserviços, os serviços estarão independentes e sua manutenção pode ser realizada sem que necessite modificar a estrutura das classes de outro serviço. Esses serviços podem ser comunicados através de um barramento, como RabbitMQ e com a utilização de um protocolo de comunicação, como AMQP ou HTTP. [[1]](https://smartbear.com/learn/api-design/what-are-microservices/)

# 1.2. Porque utilizar Arquitetura de Microsserviços

Ao utilizar a arquitetura de microsserviços, torna-se fácil gerenciar as várias funcionalidades independentes do sistema. Troca de água, verificação de potencial Hidrogeniônico, controle de temperatura, controle de luminosidade e demais, são funções autônomas entre si. Em uma aplicação monolítica, qualquer alteração feita, por menor que seja, sobre um destes serviços, implica na atualização do sistema como um todo. E em caso de falhas, estas podem se propagar para demais serviços que não necessariamente estão envolvidas na atualização realizada.

Torna-se possível, também, subdividir equipes para atuarem de maneira autônoma sobre os microsserviços, sem que haja a preocupação de afetar demais subequipes. Assim, há uma distribuição de recursos (sensores, por exemplo) mais eficiente.

# 

# 

# 

# 

# 

# 1.3. Microsserviços vs SOA

## 1.3.1. Diferenças arquiteturais

Muitos desenvolvedores realizam uma confusão por achar que a arquitetura SOA (Software Oriented-Architecture) é a mesma coisa que o estilo arquitetural de microsserviços. As duas propostas arquiteturais propõe o desenvolvimento desacoplado das funcionalidades do projeto, porém enquanto o SOA precisa de componentes que centralizam os serviços, a arquitetura de microsserviços contém os componentes descentralizados.

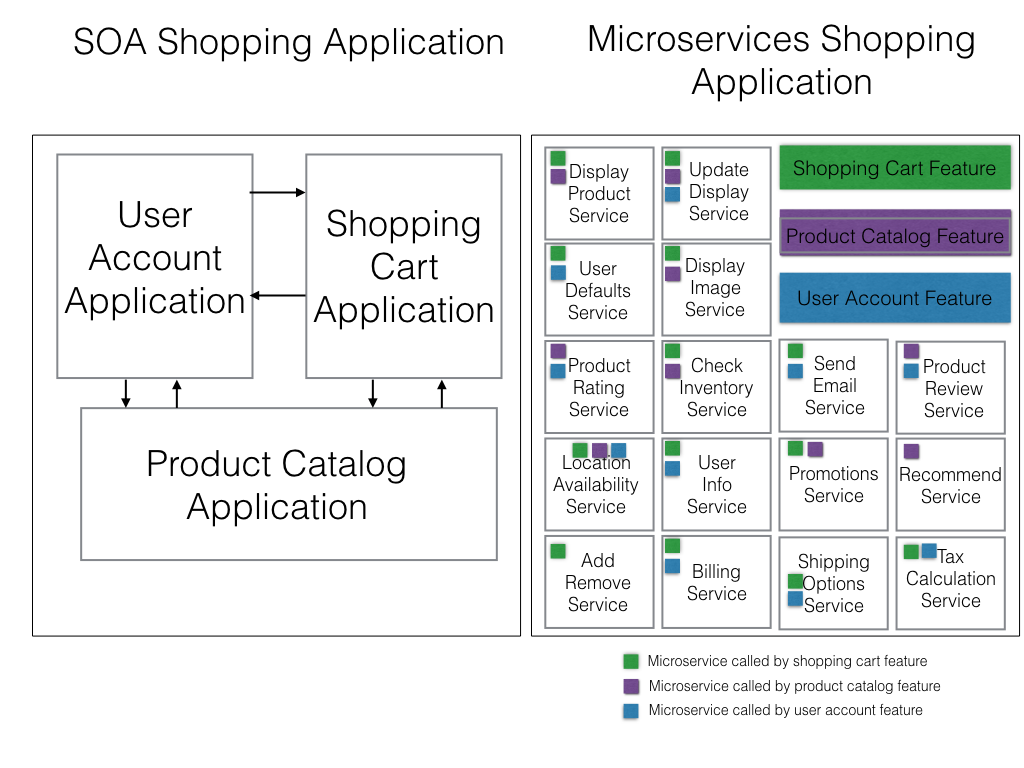
A imagem abaixo, ilustra essa diferença:

Figura 1. Exemplo de diferenças arquiteturais em um contexto de negócio de um sistema de supermercado.

Essa tabela descreve as principais diferenças entre as arquiteturas:

|  |  |  |
| --- | --- | --- |
| **Característica** | **Arquitetura SOA** | **Arquitetura de Microsserviços** |
| Tamanho do componente | Grande parte da lógica do negócio | Pequeno ou pequena parte da lógica do negócio |
| Acoplamento | Geralmente com baixo acoplamento | Sempre com baixo acoplamento |
| Estrutura Organizacional | Qualquer | Pequenas equipes multifuncionais dedicadas |
| Governança | Foco na governança centralizada | Foco na governança descentralizada |
| Objetivos | Garante que os aplicativos podem interoperar | Desenvolve novos recursos rapidamente para as organizações de desenvolvimento |

Tabela 1. Principais diferenças entre arquitetura SOa e arquitetura de microsserviços.

**1.4. Justificativa**

Baseado nas características das arquitetura SOA e microsserviços, e levando em conta as características dos *stakeholders* e do projeto em si, a arquitetura de microsserviços é mais adequada pelos seguintes fatores: facilidade na integração; utilização de *frameworks* específicos que garante a comunicação efetiva entre os subsistemas pertinentes do projeto; pelo fato de que tanto a equipe de software como a equipe de eletrônica desenvolva em qualquer microsserviços, permitindo então a governança descentralizada que a disciplina de projeto integrador 2 disponibiliza; e, a flexibilidade e pela escalabilidade que a mesma propõe.

# 

# 

# 2. Solução Proposta

Para realizar o controle e monitoramento dos dados da estufa, serão desenvolvidos sistema de software que possa ser utilizado remotamente por meio de um computador ou smartphone. Esta seção visa apresentar os requisitos deste sistema, bem como sua arquitetura.

# 2.1. Histórias de Usuário

Seguindo as recomendações do SCRUM, uma metodologia ágil de desenvolvimento de software, foram definidas histórias de usuário que representam funcionalidades planejadas do sistema. Cada história foi analisada pela equipe de desenvolvimento e pontuada segundo seu nível de complexidade. A seguir encontra-se a listagem delas.

US01 - Visualizar medições

**Pontuação:** 3

**Eu como:** Usuário

**Desejo:** visualizar as medições feitas pelos sensores

**Para que:** possa ter um monitoramento sobre a hortaliça plantada

* Exibir valores atuais de temperatura, potencial hidrogeniônico, nível de água, consumo de água, luminosidade.
* Exibir valores históricos
  + Gráficos de linha
  + Exibir dados referentes ao dia, semana, mês, ano
  + Utilizar cores diferentes para cada grandeza

US02 - Login de usuário

**Pontuação:** 5

**Eu como:** Usuário

**Desejo:** logar no sistema

**Para que:** tenha acesso a modificações no ambiente interno da estufa

* Exibir tela de login
* Utilizar endereço de email para login
* Utilizar PIN para autenticação
  + PIN de 4 dígitos

US03 - Cadastro de usuário

**Pontuação:** 3

**Eu como:** Usuário

**Desejo:** me cadastrar no sistema

**Para que:** ganhe acesso autenticado ao sistema.

* Utilizar número de série para cadastro inicial pelo aplicativo
* Cadastrar endereço de e-mail como login
  + Enviar confirmação de cadastro como e-mail
* Cadastrar PIN para autenticação
  + PIN de 4 dígitos

US04 - Manter hortaliças

**Pontuação: 3**

**Eu como:** Usuário

**Desejo:** manter uma hortaliça no banco de dados

**Para que:** diferentes tipos de hortaliças possam ser mantidos de forma automatizada pela estufa

* Cadastrar uma nova hortaliça
  + Temperatura
  + Luminosidade
  + Nome
  + Tempo de cultivo
* Excluir uma hortaliça
  + Exibir confirmação de exclusão
* Editar uma hortaliça
* Visualizar uma hortaliça

US05 - Alertar usuário

**Pontuação:** 2

**Eu como:** Sistema

**Desejo:** alertar o usuário sobre mudanças no ambiente interno da estufa

**Para que:** ele possa fazer as alterações necessárias

* Exibir um alerta quando a temperatura estiver elevada ou abaixo do necessário

US06 - Visualizar ambiente interno

**Pontuação:** 13

**Eu como:** Usuário

**Desejo:** visualizar o ambiente interno da estufa

**Para que:** possa ter um acompanhamento sobre a hortaliça plantada

* Mostrar a estufa em tempo real

US07 - Fornecer configurações para cultivo

**Pontuação:** 5

**Eu como:** Sistema

**Desejo:** fornecer configurações pré-definidas para cultivo

**Para que:** o usuário tenha um padrão para o cultivo

* Exibir um cadastro de hortaliças pré-cadastradas e hortaliças inseridas

US08 - Alterar temperatura interna

**Pontuação:** 5

**Eu como:** Usuário

**Desejo:** alterar a temperatura interna da estufa

**Para que:** seja propiciado um melhor ambiente de cultivo para a hortaliça

* Exibir um painel de controle para acionar os exaustores

US09 - Troca de água

**Pontuação:** 5

**Eu como:** Usuário

**Desejo:** acionar a troca da água do sistema

**Para que:** os nutrientes sejam renovados e a água com pH alterado seja descartada

* Exibir um painel de controle para acionar a troca da água do sistema

US10 - Visualizar relatório de volume d’água

**Pontuação:** 3

**Eu como:** Usuário

**Desejo:** visualizar um relatório com o volume de água utilizado

**Para que:** tenha controle sobre a quantidade de água gasta

* Exibir a quantidade de água utilizada

US11 - Alerta sobre falhas de comunicação

**Pontuação:** 5

**Eu como:** Sistema

**Desejo:** alertar o usuário sobre falhas de comunicação com a estufa

**Para que:** o usuário fique ciente e possa tomar providências

* Identificar origem da falha

US12 - Abrir a porta

**Eu como:** Usuário

**Desejo:** abrir a porta da estufa

**Para que:** realizar coleta da planta

Total: 52 pontos

# 2.2. Microsserviços utilizados no projeto

Para o escopo da integração e das funcionalidades pertinentes ao sistema como um todo, foram definidos os seguintes microsserviços a serem desenvolvidos pela equipe:

* Monitoramento de temperatura, e do nível hidrogeniônico da água.
* Controle de temperatura ambiente.
* Login do usuário.
* Gerenciamento de usuário.

# 2.3. Diagramas

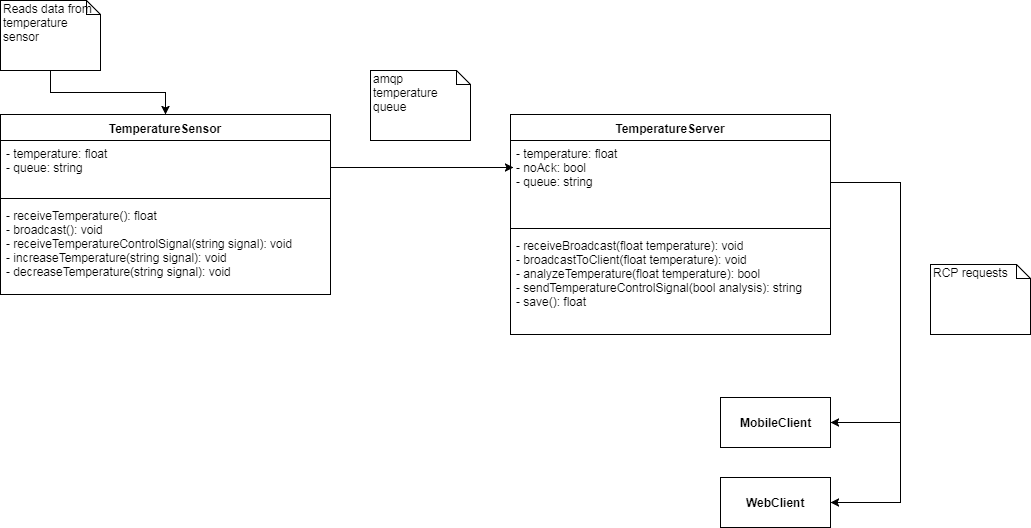
## 2.3.1. Diagrama de Classe

Os diagramas de classe representa a estrutura do sistema, em suma suas classes, atributos, métodos e relacionamentos.

Como a arquitetura de microsserviços prega o desacoplamento dos microsserviços, portanto, para cada microsserviço do projeto existe um diagrama pertinente àquele microsserviço em questão.

### Microsserviço de controle de temperatura ambiente

Na imagem abaixo, é possível observar como é a estrutura padrão para os microserviços que contam com a integração dos softwares (embarcados, webservice, cliente) como: monitoramento de temperatura, e do nível hidrogeniônico da água; e, controle de temperatura ambiente;

Figura 2. Diagrama de classe do microsserviço de controle de temperatura ambiente

### 

### Microsserviço de gerenciamento do usuário

Logo abaixo, há uma ilustração da estrutura das classes que contam com o microsserviço de gerenciamento do usuário:

## Figura 3. Diagrama de classe do microsserviço de gerenciamento do usuário

## Para o microsserviço de Login de usuário, o diagrama seguirá a mesma estrutura do diagrama acima.

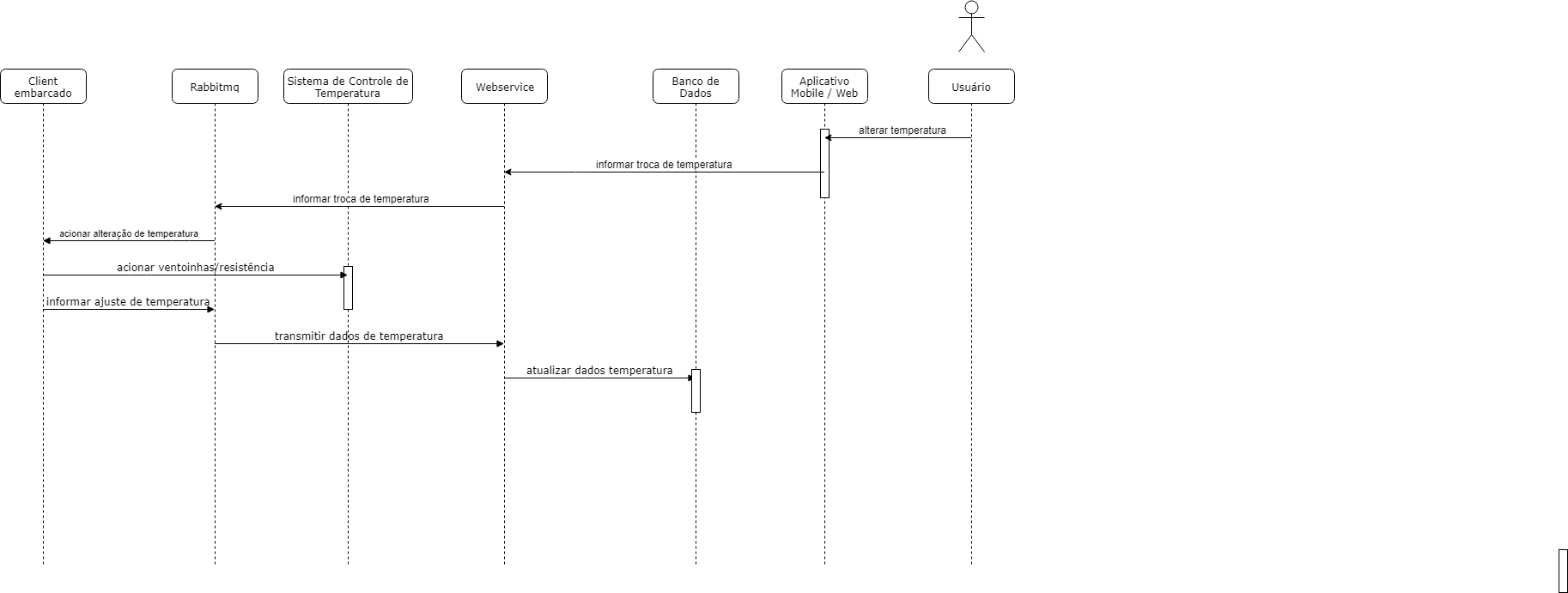
## 2.3.2. Diagrama de Sequência

Diagramas de sequência podem ser utilizados para visualizar e avaliar o fluxo lógico de um sistema. Eles exibem a forma pela qual informações, ações e eventos são transmitidos entre todas as entidades relacionadas a uma funcionalidade específica.

A seguir, são apresentados os diagramas de sequência de algumas das funcionalidades chave do projeto Greenhouse.

**Ajuste de Temperatura Manual**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas tanto pelo usuário quanto pelos sistemas para realizar o ajuste manual da temperatura.

Figura 4. Diagrama de sequência do ajuste de temperatura manual

# 

**Ajuste de Temperatura Automático**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas tanto pelo usuário quanto pelos sistemas para realizar o ajuste automático da temperatura.

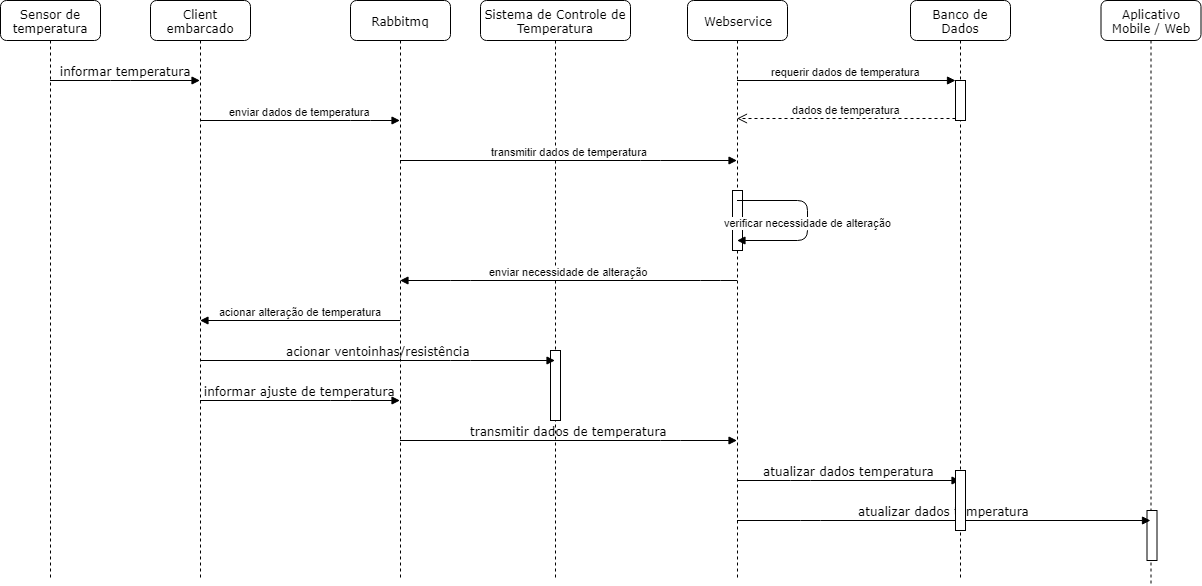
****

Figura 5. Diagrama de sequência do ajuste de temperatura automático

**CRUD Hortaliças**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas tanto pelo usuário quanto pelos sistemas para realizar as ações presentes de criar, alterar ou deletar informações de hortaliças no banco de dados.

# 

# Figura 6. Diagrama de sequência do CRUD Hortaliças

# 

**Ajuste de Iluminação**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas pelos sistemas para realizar o ajuste automático de iluminação.

# 

Figura 7. Diagrama de sequência do ajuste de iluminação

**Editar Usuário**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas tanto pelo usuário quanto pelos sistemas para realizar a edição de um usuário no banco de dados.

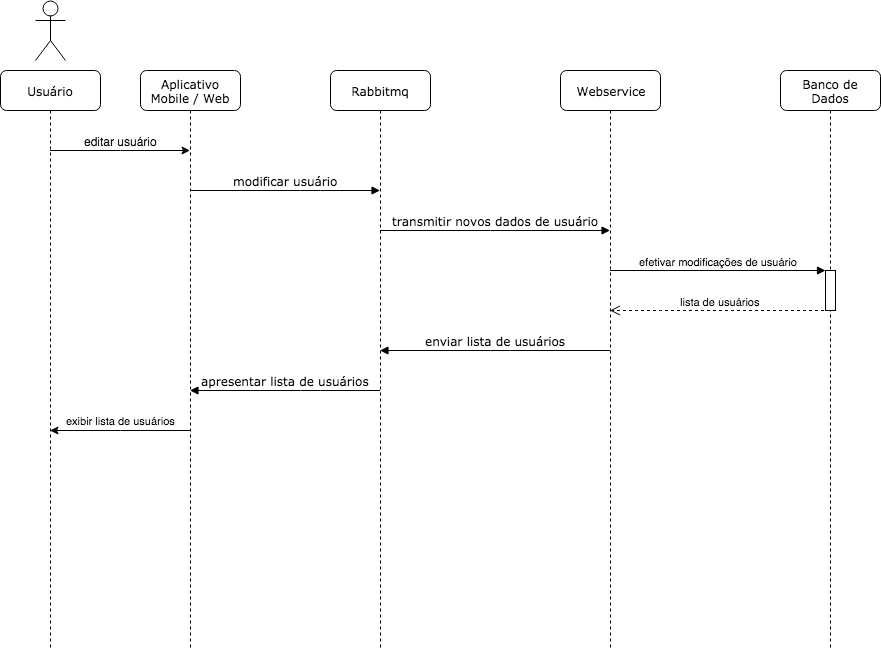
****

Figura 8. Diagrama de sequência de edição de usuário

**Enviar Alerta**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas pelos sistemas para realizar o envio de alertas ao usuário.

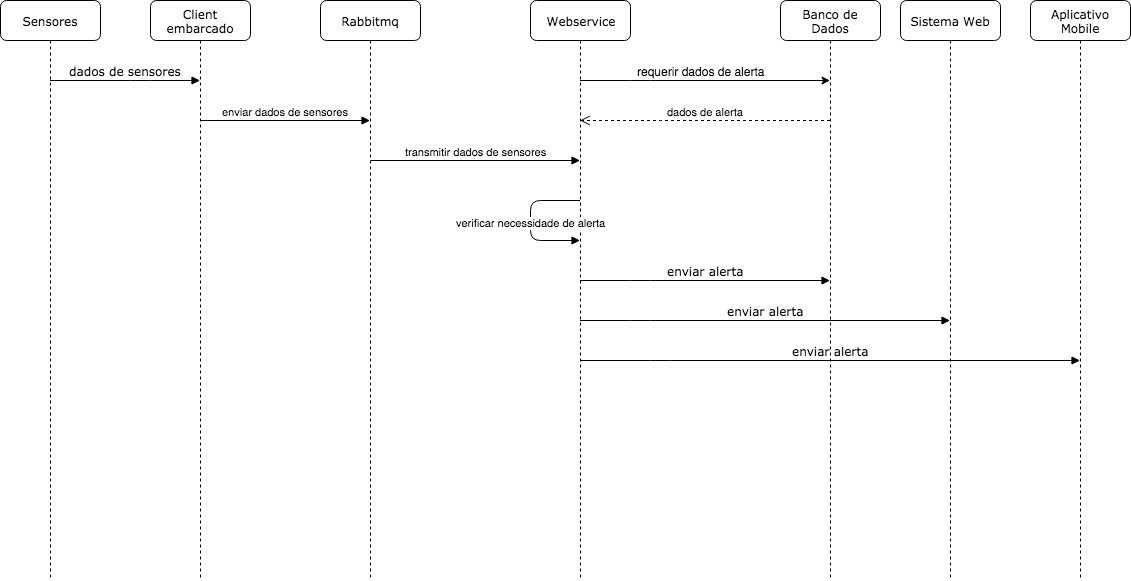
****

Figura 9. Diagrama de sequência de enviar alerta

**Exibir Relatório**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas tanto pelos sistemas para exibir o relatório das informações do ambiente da estufa (i.e. média de temperatura, umidade, iluminação, e volume de água utilizado) ao usuário.

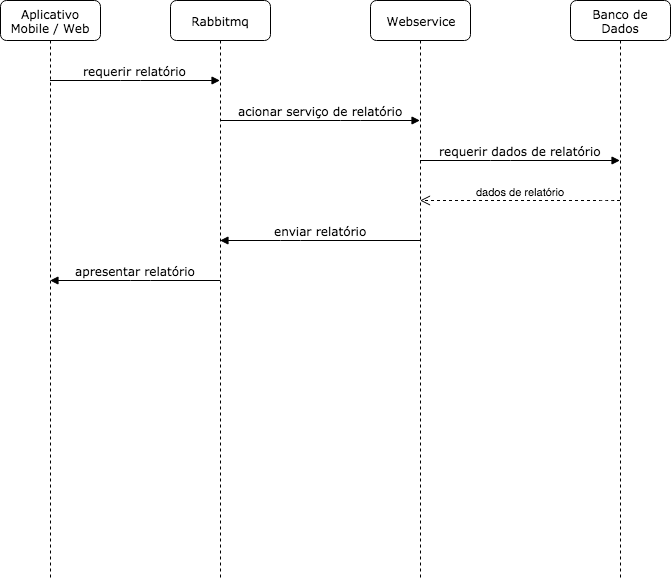
****

Figura 10. Diagrama de sequência de exibir relatório

**Troca de Água Automática**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas pelos sistemas para realizar a troca automática de água.

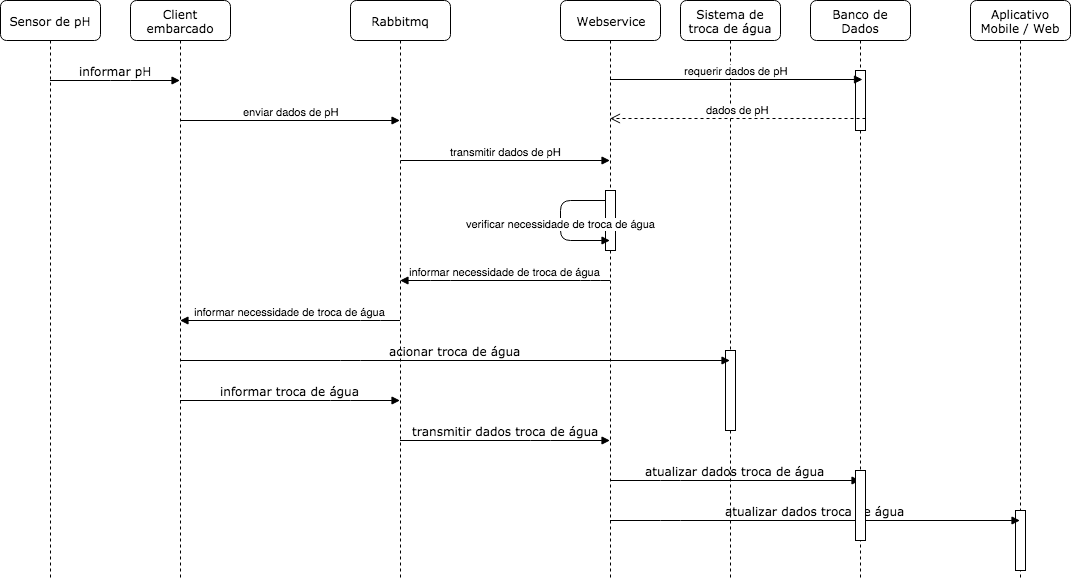
****

Figura 11. Diagrama de sequência de troca de água automática

**Troca de Água Manual**

O fluxo do diagrama de sequência abaixo mostra as ações efetuadas pelos sistemas para realizar a troca manual de água.

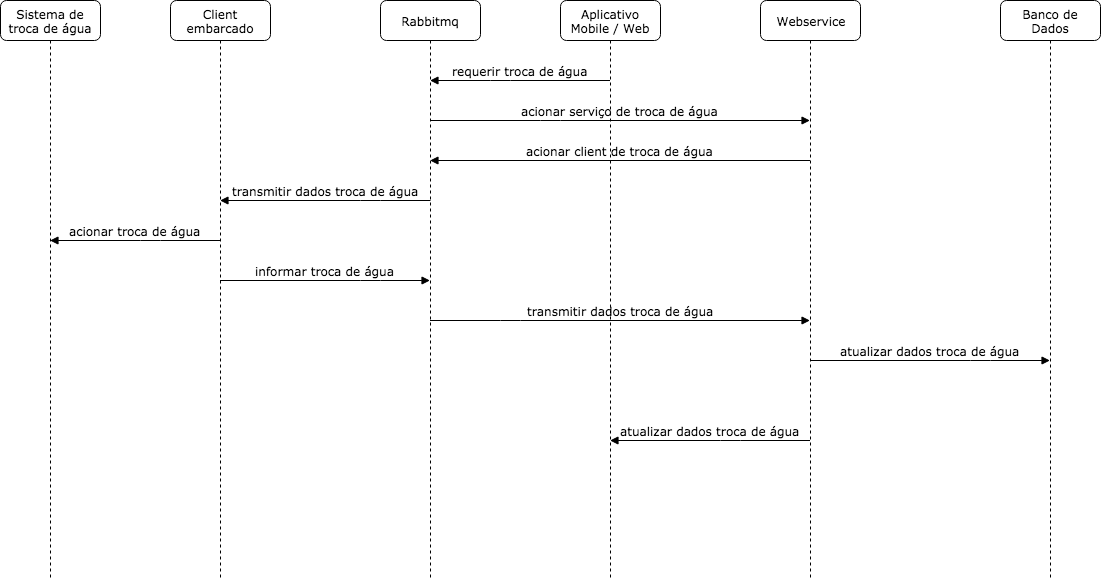
****

Figura 12. Diagrama de sequência de troca de água manual

# 

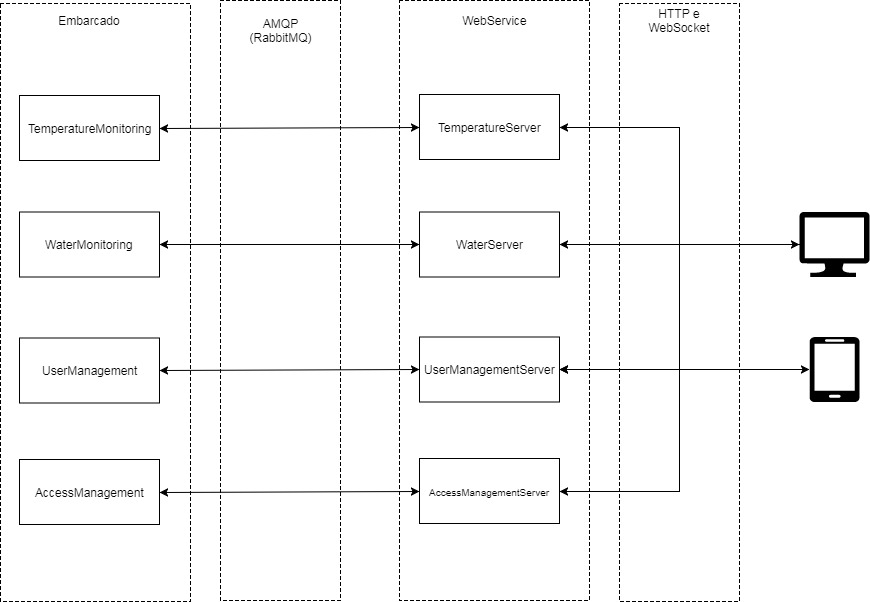
# 

# 2.4. Comunicação entre os hardwares

## 2.4.1. Raspberry PI

A comunicação entre a raspberry pi vai ser feita através de um servidor utilizando um protocolo AMQP, implementado através do RabbitMQ, que consiste no princípio de enviar uma mensagem carregando informações de um ponto a outro.

## 2.4.2. Representação ilustrativa de integração



# Figura 6. Diagrama de integração que ilustra a comunicação dos sistemas de software embarcado, web services e sistemas interface.

# 