



**UNIVERSIDADE DE ÉVORA**

**ESCOLA DE CIÊNCIAS E TECNOLOGIA**

**DEPARTAMENTO DE INFORMÁTICA**

**Plataforma de Gestão de Culturas Hidropónicas**

**Domingos Manuel Baptista de Castro Martins**

Orientação: Vitor Manuel Beires Pinto Nogueira

Co-Orientação: Luís Miguel Mendonça Rato

**Mestrado em Engenharia Informática**

Dissertação

Évora, 2014





**UNIVERSIDADE DE ÉVORA**

**ESCOLA DE CIÊNCIAS E TECNOLOGIA**

**DEPARTAMENTO DE INFORMÁTICA**

**Plataforma de Gestão de Culturas Hidropónicas**

**Domingos Manuel Baptista de Castro Martins**

Orientação: Vitor Manuel Beires Pinto Nogueira

Co-Orientação: Luís Miguel Mendonça Rato

**Mestrado em Engenharia Informática**

Dissertação

Évora, 2014



# Sumário

Este trabalho apresenta o desenho de uma plataforma de gestão e monitorização de culturas hidropónicas. Para o seu desenvolvimento foi feito um levantamento das principais definições sobre hidroponia e sistemas de monitorização. A plataforma é desenvolvida com linguagens web, para ser acessível em qualquer dispositivo móvel ou fixo com acesso à Internet. Esta possibilita a comunicação com qualquer dispositivo de controlo, seja ele amador ou industrial. A comunicação é feita através de um serviço *Representational State Transfer (REST)* disponibilizado pela plataforma, o que permite aos dispositivos de controlo alimentarem a plataforma com dados de culturas hidropónicas. Os dados recolhidos, além da utilidade para os utilizadores da plataforma, servirão também para apoiar em estudos e no desenvolvimento de melhores técnicas de manejo hidropónico.

A plataforma disponibiliza também uma espécie de rede social, que os utilizadores podem utilizar para criarem contactos, trocarem opiniões e partilharem a evolução das suas culturas.



## *Vegetable Hydroponics Management System*

# Abstract

This thesis presents the drawing of a management platform for hydroponic cultures. For its development it was made a survey of hydroponic main definitions and monitoring systems. The platform is developed with web languages, to be accessible from any mobile device or desktops with internet access. This allows communication with any control device, whether it is amateur or industrial. The communication is made through a **REST** service, which is available by the platform, that allows all the control devices to feed the platform with hydroponic cultures data. The collected data, besides the usefulness to the platform users, will also be useful to support studies and in the development of improved techniques of hydroponic management.

The platform offers a sort of social network, in which the users can make contacts, change opinions and share the evolution of their cultures.



*Dedico esta tese inteiramente aos meus pais que ao longo da minha vida, com bastante esforço, me conseguiram proporcionar o necessário para prosseguir sempre os meus estudos.*



# Agradecimentos

Agradeço aos meus pais por tudo o que, com muito esforço, me conseguiram proporcionar ao longo da vida. À Inês Pinto, pelo apoio pela ajuda ao longo destes meses. Aos meus orientadores pela disponibilidade e apoio ao longo desta dissertação. Ainda ao João Martins e Vera Tira-picos pela ajuda que me deram.



# Acrónimos

**EC** *Eletrocondutividade*

**NFT** *Nutrient film technique*

**DWC** *Deep water culture*

**PLC** *Programmable Logic Controller*

**SMS** *Short Message Service*

**MVC** *Model view controller*

**REST** *Representational State Transfer*

**IO** *Input/Output*

**W3C** *World Wide Web Consortium*

**WHATWG** *Web Hypertext Application Technology Working Group*

**EYSs** *Executive Information Systems*

**API** *Application Programming Interface*

**JSON** *JavaScript Object Notation*

**AJAX** *Asynchronous JavaScript and XML*

**SPA** *Single Page Application*

**DIY** *Do It Yourself*

**CRUD** *Create, Read, Update, Delete*

**Led** *Light Emitting Diode*



# Conteúdo

<b>Sumário</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>Lista de Conteúdo</b>	<b>xii</b>
<b>Lista de Figuras</b>	<b>xiii</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Estado da Arte</b>	<b>3</b>
2.1 Hidroponia . . . . .	3
2.1.1 Vantagens da hidroponia . . . . .	4
2.1.2 Principais sistemas existentes . . . . .	6
2.1.3 Hidroponia em Portugal . . . . .	9
2.2 Controlo . . . . .	10
2.3 Monitorização . . . . .	11
2.4 Dashboards . . . . .	11
2.5 HTML5 . . . . .	13
2.5.1 Novas funcionalidades do HTML5 . . . . .	13
2.5.2 Vantagens de utilização . . . . .	14
2.6 AngularJS . . . . .	15
<b>3 Sistema proposto</b>	<b>17</b>
3.1 Descrição . . . . .	17
3.1.1 Falhas do mercado . . . . .	18
3.1.2 Proposta de Valor . . . . .	18
3.2 Arquitetura . . . . .	19
3.2.1 Requisitos não funcionais . . . . .	21
3.2.2 Principais funcionalidades . . . . .	22
3.3 Design . . . . .	26
3.4 Usabilidade . . . . .	26
3.5 Testes . . . . .	27
<b>4 Conclusões</b>	<b>31</b>
4.1 Conclusão . . . . .	31

4.2 Trabalho futuro . . . . .	32
<b>A Estudos de interface</b>	<b>39</b>
<b>B Plataforma testes</b>	<b>49</b>
<b>C Código de um sistema</b>	<b>53</b>
C.1 Código Arduino do sistema . . . . .	53
C.2 Código python de interface . . . . .	54

# Lista de Figuras

2.1	Sistema NFT vertical . . . . .	7
2.2	Sistemas DWC . . . . .	7
2.3	Sistemas aeropônicos . . . . .	8
2.4	Sistema de gota-gota em lâ-rocha . . . . .	9
2.5	Tabela vs Gráfico . . . . .	12
2.6	HTML5CSS3JS . . . . .	14
3.1	ArquiteturaSistema . . . . .	19
3.2	SPAArq . . . . .	20
3.3	Diagram . . . . .	28
3.4	Diagram . . . . .	29



# Capítulo 1

## Introdução

Nos dias de hoje, com a crise atual que se vive, em que os produtos estão cada vez mais caros, a alimentação é um aspecto que preocupa todas as famílias. Assim a hidroponia, sendo uma técnica inovadora, que traz enormes poupanças (principalmente em água e espaço), pode ser uma opção para quem gosta de produzir os seus próprios produtos e por falta de tempo ou espaço próprio para tal, não o fazia. A hidroponia permite produzir hortícolas em praticamente qualquer lugar. Os grande problemas desta técnica são a falta de informação, o elevado conhecimento que é necessário ter sobre hidroponia e uma monitorização constante do sistema hidropónico. Assim, a construção de uma plataforma capaz de gerir e monitorizar sistemas hidropónicos, pode permitir aumentar o número de pessoas que podem ter o seu próprio sistema em casa.

Ao mesmo tempo, será possível utilizar os dados recolhidos dos utilizadores da plataforma, para contribuir para estudos sobre hidroponia e fomentar o avanço desta técnica de cultivo.

A tese está dividida em 3 partes. Na primeira parte é feito um estudo sobre hidroponia e as suas técnicas e ainda várias tecnologias que são utilizadas na plataforma. Na segunda parte é feita uma descrição da plataforma e as principais funcionalidades que são oferecidas aos utilizadores. Finalmente são tiradas algumas conclusões e apresentadas algumas sugestões para trabalho futuro.



# Capítulo 2

## Estado da Arte

De modo a permitir uma melhor compreensão da plataforma proposta neste trabalho, neste capítulo irei fazer um enquadramento teórico da área em que o trabalho se insere. Na secção 2.1 é descrito o conceito de hidroponia, as suas vantagens (2.1.1), assim como os sistemas hidropónicos existentes (2.1.2), e a evolução da hidroponia em Portugal (2.1.3). De seguida são abordados os conceitos de controlo, monitorização e dashboards.

### 2.1 Hidroponia

Trata-se de uma técnica de cultivo de plantas sem o uso de solo, onde o solo é substituído por uma solução aquosa contendo os elementos minerais indispensáveis às plantas (solução nutritiva). Este processo de cultivo está em grande expansão por todo o mundo, apresentando inúmeras vantagens em relação ao cultivo tradicional no solo.

Em alguns casos são usados substratos inertes para sustentar as raízes das plantas, como é o caso da fibra de coco ou lã-rocha.

O maneiio de sistemas hidropónicos tem muitas variáveis a ter em conta, sendo este o principal fator que impede novos produtores, com poucos conhecimentos na área, de se dedicarem a esta técnica.

Nestes sistemas há que ter em conta o pH e *Eletrocondutividade (EC)*. No decorrer do ciclo de produção, as plantas vão consumindo os nutrientes; a partir da monitorização dos valores de *EC* é possível saber quando é necessário ajustar ou renovar a solução de nutrientes. Quanto ao pH é fundamental mantê-lo dentro de um determinado intervalo de valores (conforme a cultura), para que as plantas absorvam os nutrientes da melhor forma. A humidade e a temperatura ambiente da solução são também fatores importantes que

exigem monitorização constante.

Na hidroponia é vital manter uma constante vigilância do sistema e estar atento às alterações nas plantas, daí a importância de sistemas que ajudem a monitorizar todos estes fatores, de forma a aumentar a produtividade dos sistemas.

### 2.1.1 Vantagens da hidroponia

De seguida observam-se as vantagens da utilização de culturas hidropónicas em comparação com as culturas no solo. De salientar a poupança da água, que segundo [?] num sistema comercial pode chegar a  $1/20$  do sistema tradicional para a mesma quantidade produzida.

As elevadas produtividades aliadas à qualidade e disponibilidade dos produtos fora de época, fazem com que os sistemas hidropónicos sejam altamente promissores e especialmente apetecíveis em países onde a falta de espaço ou água não permitem a utilização das culturas tradicionais. Em países onde a luz não é constante, como é o caso dos pólos (onde em certas alturas do ano é sempre noite), podem ser aplicadas luzes artificiais, para continuar a produzir. Com o aparecimento dos Led's *Light Emmitting Diode (Led)*, embora ainda seja uma tecnologia relativamente cara, possibilita com baixos custos de energia produzir nestes países ou em locais fechados (caves - com temperaturas estáveis durante todo o ano).

### Vantagens das culturas em terra vs culturas hidropónicas

Adaptação da tabela retirada de: [6]

#### Esterilização

**(Solo) S** - Vapor, químicos, requer bastante tempo (2 a 3 semanas).

**(Hidroponia) H** - Vapor, químicos, requer pouco tempo.

#### Nutrição das plantas

**S** - Muito variável; deficiências localizadas; frequentemente indisponível devido a fraca estrutura do solo, difícil de controlar, monitorizar e ajustar.

**H** - Controlado; relativamente estável; homogéneo para todas as plantas; fácil de controlar, monitorizar e ajustar.

#### Espaço de plantação

**S** - Limitado pela nutrição do solo e luz.

**H** - Limitado apenas pela luz, fazendo com que seja possível utilizar o espaço todo de forma a produzir mais unidades por área.

### **Controlo de plantas daninhas**

**S** - Presença de plantas daninhas e necessidade de as retirar frequentemente.

**H** - Ausência de ervas daninhas.

### **Doenças e pragas**

**S** - Presença de vários insetos, doenças e animais que podem atacar as plantas; uso frequente da rotação de plantio para evitar infestações.

**H** - Sem doenças, insetos e animais no substrato; Sem necessidade de fazer rotação de plantio.

### **Água**

**S** - As plantas são bastante sujeitas ao stress de água devido à fraca estrutura do solo; Não podem ser usadas águas salinas; Uso ineficiente da água, a maior parte é perdida.

**H** - As plantas não sofrem stress de água; podem ser usados mecanismos automáticos para a gestão da água; Uso bastante eficiente da água, sem perdas significativas.

### **Qualidade do produto**

**S** - Frequentemente os frutos são moles ou inchados devido à falta de potássio e cálcio, resultando numa menor vida na prateleira.

**H** - A fruta é firme e tem uma longa vida de prateleira. Isto permite que seja expedida para longas distâncias.

### **Fertilizantes**

**S** - Uso de grandes quantidades, sendo que 50% a 80% são perdidos por se infiltrarem para fora da zona das raízes.

**H** - Uso de pequenas quantidades, distribuição uniforme por todas as plantas.

### **Higiene pública**

**S** - Os desperdícios orgânicos usados na agricultura do solo causam problemas de saúde aos humanos.

**H** - Não se adicionam agentes biológicos aos nutrientes; Sem doenças humanas registadas.

### **Transplantação**

**S** - Necessidade de preparar o solo; frequentemente acontecem choques de transplantação, que retardam o crescimento ou matam as plantas; Difícil de controlar os parâmetros do solo

**H** - Sem necessidade de preparação do meio, choque de transplante bastante reduzido, fácil de controlar solução de nutrientes.

### **Maturação**

**S** - Frequentemente lenta devido às fracas condições do solo.

**H** - Com as condições de luz adequadas as plantas têm um período de maturação bastante mais rápido do que no solo.

**Permanência**

**S** - O solo das estufas precisa de ser mudado frequentemente (de vários em vários anos), pois perde as capacidades de fertilidade.

**H** - Sem necessidade de pousio. Os substratos inertes utilizados podem durar vários anos se forem corretamente esterilizados.

**Rendimento**

**S** - Tomates de estufa, 7 a 9 kg de fruto por pé.

**H** - Tomates de estufa, 22 a 31 kg de fruto por pé.

**2.1.2 Principais sistemas existentes**

Atualmente alguns dos sistemas utilizados em hidroponia são: *Nutrient film technique* (**NFT**), Aeropónico, DWC, Ebb and flow e Gota-gota, sendo utilizados substratos inertes (argila expandida, fibra de coco, areia, gravilha, lâ-rocha) para suportar a planta. Segundo [6] os sistemas mais utilizados mundialmente são o **NFT** e o Gota-gota com utilização de lâ-rocha como substrato.

**NFT**

O **NFT** é uma técnica de hidroponia onde as plantas são inseridas em calhas ou tubos, estando as suas raízes em contacto com uma película da solução de nutrientes, que circula em circuito fechado por todo o sistema.

Estes sistemas são muito sensíveis a falhas de energia, pois se a bomba que faz a circulação da solução não funcionar, as plantas podem morrer em pouco tempo. O sistema **NFT** é bastante utilizado tanto em sistemas caseiros, como em sistemas comerciais, pois produz bons resultados e não tem uma manutenção e controlo muito difícil. Em casa este sistema pode ser utilizado na vertical (figura: 2.1), de forma a rentabilizar uma parede ou um espaço pequeno como uma varanda, nos sistemas comerciais habitualmente são utilizados na horizontal dentro de estufas de forma a aproveitar ao máximo a luz natural existente.

Em Portugal, a maioria das explorações hidropónicas, utilizam este sistema, sendo poucas as que utilizam outros métodos.

Um dos grandes problemas desta técnica é a temperatura da solução, que durante o verão, pelo facto da solução circular numa fina camada dentro das calhas, torna-se difícil de controlar.

A desinfecção do sistema é facilmente conseguida, pois basta isolar a parte do sistema pretendido (através do fecho de válvulas de circulação) e proceder à limpeza. Este é um fator bastante importante para evitar a propagação de doenças por todo o sistema.

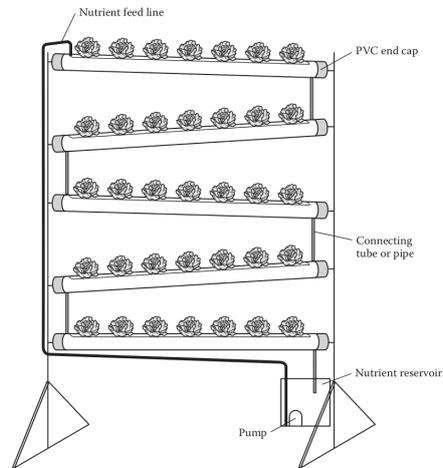


Figura 2.1: Sistema NFT vertical (Fonte [6])



(a) Raft system



(b) Bucket system

Figura 2.2: Sistemas DWC (Fonte [4])

### DWC - Flutuante

Neste sistema existem várias abordagens, como se pode verificar nas imagens 2.2.

Existe um sistema de jangada, onde as plantas são colocadas em orifícios nas placas, que flutuam num tanque onde está contida a solução (figura 2.2a). Neste sistema a maior vantagem é a facilidade em manter a temperatura e os parâmetros de EC e PH da solução, pois como a quantidade de solução é maior, é mais difícil haver uma mudança brusca nas condições. É importante haver uma circulação na água de forma a injetar o oxigénio necessário às das plantas.

Por outro lado temos um sistema de baldes (Figura 2.2b), mais utilizado em sistemas caseiros, devido à possibilidade de fazer um sistema com apenas um balde ou caixa de plástico, que possibilita colocar o sistema em locais mais apertados. Este sistema também é utilizado em culturas comerciais, sendo bastante utilizado em tomateiros devido ao grande porte e às raízes bastante desenvolvidas.



(a) Exemplo de um sistema aeropónico

(b) Detalhe de um sistema aeropónico

(c) Bucket system

Figura 2.3: Sistemas aeropónicos (Fonte [5])

### Aeropónico

O sistema Aeropónico é provavelmente o mais avançado tecnologicamente e o mais difícil de construir. As raízes ficam suspensas no ar e são borrifadas com a solução nutritiva. A nebulização é feita geralmente a cada dois minutos. Uma vez que as raízes são expostas ao ar, como no sistema NFT, as raízes têm o mesmo problema de secar rapidamente, caso os ciclos de nebulização sejam interrompidos ou feitos em longos períodos de tempo.

Nos últimos anos têm-se feito experiências com este sistema para a produção de batatas, onde as raízes estão numa caixa grande e escura, com uma porta lateral para recolher as batatas. Neste sistema a produção é bastante superior ao cultivo tradicional.

### Ebb and flow

Esta técnica é conhecida por ser bastante simples. Consiste numa caixa de cultivo cheia com um substrato inerte (normalmente argila expandida ou gravilha), onde as plantas ficam seguras; por baixo da caixa de cultivo, está colocado o reservatório com os nutrientes. A solução de nutrientes é continuamente bombeada do reservatório para a caixa de cultivo, que quando fica cheia, se esvazia graças ao sifão de bell<sup>1</sup>.

As vantagens deste método assemelham-se às do Flutuante, sendo o Ebb and flow mais utilizado em sistemas caseiros de pequena dimensão.

### Gota-Gota

Este sistema é bastante utilizado para produzir plantas com um porte maior, como pimenteiros ou tomateiros. Normalmente utiliza como substrato a lâ-rocha, que é pousada em cima de calhas de forma a poder reaproveitar a solução.

<sup>1</sup>sifão de bell - <http://www.ctahr.hawaii.edu/oc/freepubs/pdf/BIO-10.pdf>



Figura 2.4: Sistema gota-gota em lâ-rocha (Fonte [30])

As vantagens deste tipo de sistema são a poupança e reaproveitamento da solução nutritiva, assim como o aumento da produção e maximização do espaço de cultivo. Este método é semelhante ao NFT mas com a vantagem de não ter problemas com a temperatura da solução.

### 2.1.3 Hidroponia em Portugal

Em Portugal, esta é uma técnica relativamente recente e difícil de aplicar, devido aos cuidados e conhecimentos necessários ao manuseio hidropónico, já que existem ainda poucas explorações comerciais implementadas no nosso país.

De salientar a Hubel<sup>2</sup>, que é uma grande produtora de frutos vermelhos hidropónicos e que exporta aproximadamente 90% da sua produção. Existem outras empresas hidropónicas que produzem alface, pimentos e outros legumes de folha (salsa, rúcula, etc).

A nível caseiro algumas pessoas começam a experimentar construir o seu próprio sistema, mas deparam-se com bastantes dificuldades em encontrar informação de fonte portuguesa e assim de manusear corretamente o seu sistema.

---

<sup>2</sup><http://www.hubel.pt/hv/>

## 2.2 Controlo

O controlo é utilizado para modificar o comportamento de um sistema de forma a este se comportar da forma desejada. O controlo tem uma história que se pode confundir com a evolução do Homem e a sua necessidade de controlar e retirar o máximo de proveito dos recursos e dos sistemas que nos rodeiam. Os métodos tradicionais, por serem em grande parte manuais, envolvem uma monitorização complicada, obrigando a deslocação ao local para verificar o estado dos sistemas.

Nos dias de hoje, o controlo faz parte das nossas vidas; podemos encontrar os mais variados sistemas de controlo em quase tudo o que nos rodeia: estão presentes desde as fábricas, em sistemas industriais, até aos gadgets e eletrodomésticos que temos nas nossas próprias casas. Com o avanço dos meios tecnológicos, é possível assistir a uma proliferação dos sistemas de controlo automático, controlados à distância através da Internet.

Um sistema de controlo é constituído essencialmente por 4 partes: os sensores, os atuadores, a rede de comunicação e o dispositivo+algoritmo de controlo. Para se construir um bom sistema de controlo é necessário existir um conhecimento global sobre todo o sistema e os seus componentes, desde os sensores e atuadores que mais se adequam até às perturbações, ruídos e incertezas que podem afetar o controlo. No caso específico da hidroponia, é preciso ter noções de hidroculutura e fito-cultura, de forma a saber quais os aspetos que precisam de ser monitorizados e controlados, para assim construir o melhor modelo matemático adequado à hidroponia.

Os sensores são a "entrada" dos sistemas, permitem recolher dados para serem enviados para o controlador através da rede de comunicação. Na maior parte das vezes, os controladores são computadores com microprocessadores capazes de executar algoritmos de controlo, que depois atuam no sistema através dos atuadores.

Os sistemas de controlo lógico são, atualmente, os mais utilizados na indústria, sendo na sua maioria implementados através de *Programmable Logic Controller (PLC)* e microcontroladores. Estes controladores podem ser ligados a vários sensores e atuadores e serem utilizados para diversas operações e aplicações. Alguns exemplos incluem elevadores, linhas de montagem, caldeiras e outros equipamentos com operação de controlo on-off automático.

Na solução estudada, a componente de controlo é independente da plataforma e pode ser de qualquer tipo, desde que possibilite a comunicação com o servidor de monitorização

através do protocolo estabelecido. Desta forma o utilizador é livre de implementar o seu próprio sistema e algoritmo de controlo, adequando-o o melhor possível ao seu sistema.

## 2.3 Monitorização

Monitorizar (significa acompanhar por meio de um monitor, fazer vigilância, supervisionar) é um comportamento que está integrado no ser humano; esta necessidade nasce do facto do homem garantir que outros que estão sob a sua alçada cumprem algo da forma desejada (por exemplo: o supervisor de uma fábrica que inspeciona o produto final de uma linha de montagem para garantir a qualidade desejada).

Transportando a monitorização para uma época mais recente, esta é muito utilizada nos meios financeiros para acompanhar os mercados e os vários valores em tempo real, de forma a tomar as melhores decisões. Para essa monitorização, são utilizados Dashboards, com variáveis, gráficos e/ou tabelas com os dados necessários para se tomarem decisões.

Uma plataforma de monitorização é normalmente constituída por vários Dashboards e relatórios que se alimentam de dados de vários locais e os relacionam para serem lidos de uma forma intuitiva e rápida pelos utilizadores.

## 2.4 Dashboards

Um dos grandes problemas quando se está a analisar dados é, na maioria das vezes, a dificuldade de os ler e interpretar, uma vez que normalmente são apresentados em tabelas ou folhas de cálculo [29], que tornam difíceis as leituras e a perceção de relações entre as várias fontes de dados.

Os Dashboards (também conhecidos como *Executive Information Systems (EYSs)*), surgiram por volta de 1980 [18] em implementações exclusivas para os executivos, com o objetivo de apresentar medidas financeiras numa interface simples que os executivos conseguissem compreender. A partir dos anos 90, o Data Warehousing e o Business Intelligence impulsionaram o uso dos Dashboards e tornaram-nos numa ferramenta de uso diário para monitorizar eficientemente os dados, aumentando desta forma o desempenho dos negócios/sistemas.

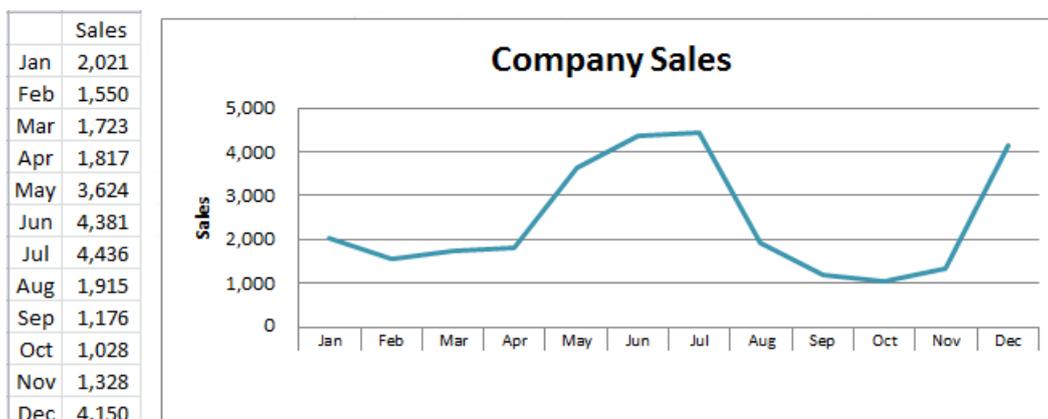


Figura 2.5: Tabela vs Gráfico (Fonte [13])

O termo Dashboard refere-se a uma simples página onde é mostrada informação pertinente para monitorizar o que se passa relativamente a algum aspeto do negócio. Em qualquer negócio ou sistema é importante monitorizar para se manter um conhecimento da situação em tempo útil. Em diversas áreas, monitorizar é extremamente importante, quer seja para aumentar o desempenho ou simplesmente para garantir que tudo está dentro do normal assegurando a qualidade/desempenho pretendida.

Como se verifica na Figura 3.2, a utilização de gráficos facilita a perceção e rapidez de leitura dos dados. Por sua vez, a leitura da tabela leva mais tempo já que fazer uma figura mental de mudança dos dados no tempo é bastante mais difícil e não fica na memória tão facilmente.

Num Dashboard nunca se deve mostrar mais informação que aquela que a pessoa que o vai utilizar é capaz de processar rapidamente. Essa quantidade de informação depende muito da forma em que é apresentada e da capacidade e experiência do utilizador.

Há que ter também em conta que um Dashboard muito "espalhafatoso" (cheio de cores e efeitos especiais, por exemplo), está longe de ser ergonómico e perceptível. A utilização de cores num Dashboard tem de ser moderada, pois caso não o seja, leva à distração e perda de foco no que é essencial.

Para se chamar a atenção para uma informação importante e que requer atenção imediata, deverá ser usada uma cor que se destaque das restantes. Além da cor, também se pode utilizar sons ou avisos flutuantes em locais visíveis ou próprios para o efeito.

No desenho dos Dashboards, deve-se deixar algo para os utilizadores fazerem, isto é, não se pode dar tudo ao utilizador, é preciso que ele se sinta encorajado a pensar sobre os dados e não apenas a reagir sobre o que lhe é apresentado, pois desse modo o utilizador vai perder a atenção e, quando necessário, não vai conseguir reagir prontamente a um problema.

## 2.5 HTML5

HTML é uma linguagem utilizada para representar e estruturar conteúdos sobre uma página web. O HTML5 é a sua versão mais recente e traz uma "lufada de ar fresco" à forma de utilização do HTML, através das suas novas funcionalidades. O HTML5 tem origem de um desenvolvimento conjunto entre *World Wide Web Consortium (W3C)* e *Web Hypertext Application Technology Working Group (WHATWG)*. As especificações do HTML 5 ainda estão em desenvolvimento, pelo que ainda existem algumas funcionalidades que não são standard. Esta nova versão veio substituir o HTML 4 que era recomendado pelo W3C desde 1997. As especificações do HTML 5 existentes começaram a ser desenvolvidas em 2004 a partir de estudos das aplicações do HTML que já existiam [33].

### 2.5.1 Novas funcionalidades do HTML5

*'HTML is still a draft'* [34]

Desde 2004, o HTML 5 tem sofrido alterações que servem para tornar standard várias funcionalidades novas, assim como discutir quais as novas funcionalidades e especificações que ficam ou desaparecem do HTML 5. Relativamente à sintaxe do HTML, podemos destacar várias novas tag's de estrutura, que melhoram a forma de como é possível estruturar as páginas web. De destacar algumas como:

- **section** - representa uma secção genérica
- **article** - representa uma entrada independente de conteúdo, como por exemplo notícias de um blog
- **header**
- **footer**
- **nav** - secção dedicada à área de navegação da página
- **video, audio** - para conteúdo multimédia
- **canvas** - utilizada para renderizar bitmaps dinâmicos
- **input** - com novos tipos: url, email, date, number, color



Figura 2.6: HTML5 - CSS3 - JS (Fonte [25])

Destes novos elementos podemos destacar o vídeo, que possibilita a visualização de vídeo sem a utilização de plugins (como Flash e Quicktime), nos browsers com suporte HTML 5. Além de acrescentar novas funcionalidades, as novas tag's também permitem uma melhor organização da sintaxe. Outra importante funcionalidade é a geolocalização, que permite criar aplicações com serviços baseados em localização e que podem tirar partido dos dispositivos de GPS dos Smartphones para criar aplicações que se comportem de forma semelhante às aplicações nativas.

A "Offline storage" é uma funcionalidade pertinente que permite armazenar dados no lado do cliente para ser possível utilizar as páginas web sem ligação à internet; quando houver ligação há uma sincronização de dados com o servidor. Isto poderá ser importante nos dispositivos móveis, pois poderá ser utilizado para guardar ficheiros e dados, reduzindo o tempo de carregamento e tráfego quando se utilizam ligações móveis de internet. O canvas é um dos novos elementos com mais importância, já que permite criar uma área onde é possível renderizar imagens vetoriais. O controlo do elemento é feito com Javascript e permite criar animação e formas de interação com o utilizador, através do rato ou teclado.

### 2.5.2 Vantagens de utilização

A principal vantagem do HTML 5 é o suporte para praticamente todas as plataformas e browsers existentes (multi-plataforma). A grande quantidade de sistemas operativos e linguagem diferentes implica que as aplicações nativas não sejam multi-plataforma, tendo de existir um enorme espaço e investimento para desenvolver para várias plataformas. Esta heterogeneidade pode ser ultrapassada desenvolvendo as aplicações em HTML 5 e Javascript, utilizando o mesmo desenvolvimento para todos os sistemas, concentrando os esforços e recursos em apenas uma linguagem.

Tendo em conta o suporte para a maioria dos vários sistemas operativos, sobram as restantes características dos dispositivos (como por exemplo a resolução, tamanho do ecrã,

se é touch). Com o CSS3 e as suas novas funcionalidades é possível garantir a correta visualização da aplicação nos vários dispositivos. Esta nova faceta do CSS3 é conseguida através de Media Queries, que permitem aplicar estilos diferentes para tamanhos de ecrãs diferentes, tamanhos de elementos relativos ou proporcionais, "overflows" e também "ward wrap". Outra vantagem de utilização do HTML 5 é a grande quantidade de frameworks e bibliotecas de Javascript, que quando utilizados em conjunto com o HTML 5 facilitam o desenvolvimento e enriquecem a experiência de utilização ao apresentar interfaces apelativas e interativas.

## 2.6 AngularJS

HTML is great for declaring static documents, but it falters when we try to use it for declaring dynamic views in web-applications. AngularJS lets you extend HTML vocabulary for your application. The resulting environment is extraordinarily expressive, readable, and quick to develop.[22]

Angular JS é uma framework de desenvolvimento Javascript, bastante flexível, que permite estender as funcionalidades do HTML. Esta framework devido ao seu dinamismo é considerada uma das mais adequadas para desenvolver aplicações *Single Page Application (SPA)*, o que a torna numa ótima escolha para esta plataforma de monitorização. A framework Angular JS foi desenvolvida pela Google em 2009, sendo inicialmente utilizada apenas no interior da empresa. Atualmente, já foi formada uma framework em open source, o que permite o aparecimento e crescimento de uma grande comunidade que desenvolve funcionalidades e módulos assentes na framework.

Por defeito, o Angular JS possui uma quantidade de ferramentas limitada, mas graças à sua flexibilidade é possível acrescentar novas ferramentas e funcionalidades, o que permite uma grande adaptação ao desenvolvimento de novas aplicações mais interativas e atrativas para os utilizadores.

A utilização desta e de outras frameworks agiliza e encurta o desenvolvimento de aplicações, uma vez que nos poupa trabalho ao fornecer uma estrutura de desenvolvimento já testada, assim como o acesso ao conhecimento das comunidades das respetivas frameworks.



# Capítulo 3

## Sistema proposto

Neste capítulo irei fazer um enquadramento teórico sobre a plataforma e a ideia. Primeiro irei descrever a ideia e o modelo de negócio associado, seguidamente a arquitetura e desenho da plataforma. Finalmente abordarei os testes efetuados.

### 3.1 Descrição

A plataforma visa tornar simples e eficaz a gestão e monitorização, possibilitando aos sistemas existentes uma entrada na era digital. Assim, o objetivo é reunir num único local uma grande comunidade de profissionais e entusiastas de hidroponia, criando simultaneamente uma base de dados sobre hidroponia única a nível mundial. A plataforma é a primeira a agregar num só local a gestão, monitorização e rede social sobre hidroponia. Trata-se portanto de um serviço baseado em novas tecnologias que visa impulsionar a utilização desta técnica de cultivo e a comunicação entre vários utilizadores, profissionais ou amadores.

A plataforma é um serviço orientado a um público específico: pessoas com alguma ligação ou interesse à área de hidroponia.

Serve sobretudo de apoio aos utilizadores amadores, mas também aos profissionais; isto é, permite que qualquer pessoa além de monitorizar e gerir o seu sistema possa trocar opiniões, estabelecer contactos e até pedir ajuda aos outros utilizadores da plataforma.

A plataforma fornece mecanismos de acesso a informação diversa (como tipos de culturas hidropónicas, artigos, notícias) permitindo ainda a utilização da rede social para partilha

de informação e criação de contactos de cariz profissional ou particular.

Os utilizadores têm a possibilidade de gerir e monitorizar os seus sistemas, assim como definir todos os parâmetros que lhes dizem respeito. É também possível criarem da forma mais adequada os seus dashboards de monitorização e partilharem informações dos seus sistemas nas redes sociais.

### 3.1.1 Falhas do mercado

Através da pesquisa efetuada, verificou-se a existência de várias lacunas na área da hidroponia, nomeadamente as seguintes:

- Serviços/sistemas de monitorização e apoio à gestão pouco flexíveis e com custos muito elevados;
- Dificuldade em centralizar a gestão e monitorização de vários sistemas num único local;
- Dispersão da informação disponível sobre hidroponia;
- Escassez de comunidades abertas unicamente sobre hidroponia (as que existem são maioritariamente no facebook e de pequena dimensão);
- Serviços de monitorização pouco intuitivos e de difícil utilização, sendo maioritariamente de uso exclusivo em computadores;
- Dificuldade em estabelecer contactos entre stakeholders;
- Pouca informação aberta sobre hidroponia;

### 3.1.2 Proposta de Valor

Esta plataforma é centrada em soluções para hidroponia: para público amador, profissional, curiosos e outras entidades com interesse no setor. A plataforma pretende tornar-se uma referência na área e ajudar no seu desenvolvimento com as ferramentas e informação adquirida. A existência de deficiências nos sistemas e plataformas atuais torna esta plataforma numa proposta necessária para a simplificação da utilização do cultivo hidroponico. A plataforma tem como objetivo ajudar os utilizadores a maximizar o proveito das suas culturas.

A rede social criada permite que os utilizadores, além de estabelecerem contactos uns com os outros, disponibilizem informação pública sobre os seus sistemas ou quaisquer outros assuntos relacionados com o tema.

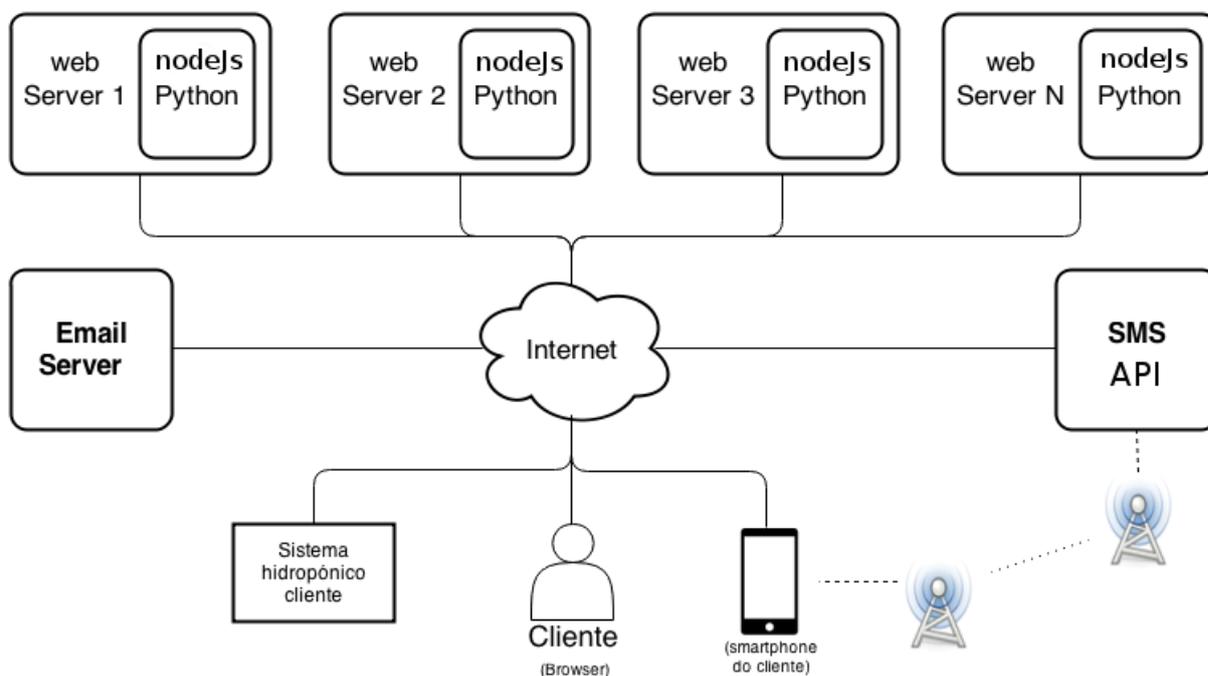


Figura 3.1: Arquitetura do sistema

## 3.2 Arquitetura

A arquitetura em que se vai basear o sistema é centrada no tipo cliente-servidor. A escolha desta arquitetura recai em vários motivos, entre eles: a possibilidade de múltiplos acessos concorrenciais, a localização dos servidores, possibilidade de aumentar a capacidade do sistema.

A conexão dos clientes é feita através da Internet, a partir de um browser para aceder a plataforma ou a partir de pedidos **REST** para a ligação aos sistemas de controlo. A base de dados consegue suportar os múltiplos acessos e escritas concorrenciais e está replicada pelos vários servidores de forma a garantir uma gestão eficiente dos dados de clientes. Os servidores de email servem para enviar dados, pedidos e alertas para os utilizadores, que podem ter origem em ações do utilizador ou parametrizações de avisos criadas para enviar um email quando um valor do sistema está fora de um determinado intervalo. Equivalente ao servidor de email existe também o servidor de sms, que é um serviço fornecido por uma entidade externa e que funciona através de uma *Application Programming Interface (API)* de ligação HTTP.

Esta arquitetura de sistema facilita a escalabilidade do sistema, pois é bastante fácil acrescentar servidores em sítios distintos e garantir um balanço de processamento equilibrado entre os vários servidores. Além disso, a possibilidade dos servidores estarem localizados em vários locais, aumenta a performance da ligação dos clientes, pois os pedidos podem ser redirecionados para os servidores mais próximos com disponibilidade de

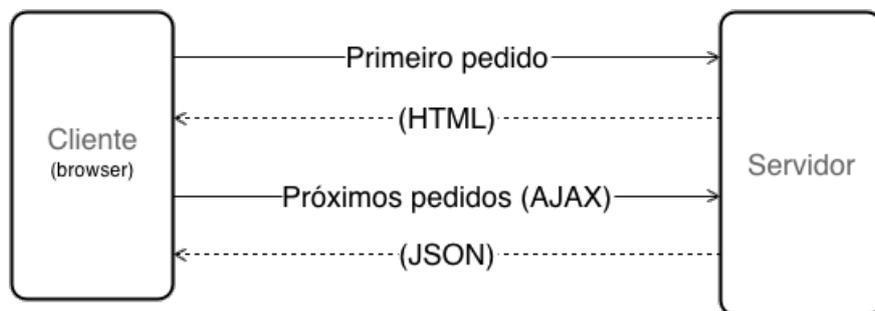


Figura 3.2: funcionamento SPA

processamento para os atender.

A plataforma funciona de uma forma distinta do que é normal no que diz respeito à arquitetura tradicional de construção das páginas de web standard, pois utiliza uma arquitetura SPA, em que no primeiro pedido do browser é retornado HTML, e os restantes pedidos são apenas pedidos *Asynchronous JavaScript and XML (AJAX)* que retornam dados *JavaScript Object Notation (JSON)*. Desta forma podemos tirar partido da **API REST**, que está implementada para fazer a ligação com os sistemas de controlo.

Para gerir e controlar todos o processamento do lado do cliente é utilizada a framework de JavaScript AngularJS. Com esta framework podemos criar uma plataforma que é suportada em grande parte por processamento do lado do cliente, isto é, o servidor retorna dados que depois são tratados e apresentados no browser pelo AngularJS. Desta forma reduzimos o processamento do lado do servidor, o que possibilita obter uma melhor performance para uma maior quantidade de clientes.

O aspeto negativo desta arquitetura é que uma grande parte do código da plataforma é exposta ao cliente em ficheiros JavaScript, o que permite que facilmente outras plataformas concorrentes possam tirar partido da lógica de programação implementada no lado do cliente. Este é um risco que se têm que correr para manter o dinamismo das páginas do lado do cliente e assim evitar gastos em tráfego com um elevado numero de pedidos ao servidor e atualizações constantes de todo o conteúdo da página.

A ligação entre os sistemas de controlo e o servidor é feita através de uma **API REST** assente em pedidos HTTP. Para tal, os clientes têm que criar uma chave na plataforma que irá servir de autenticação/identificação; esta chave é enviada no *Header* dos pedidos HTTP. Estes pedidos são acompanhados com a informação recolhida pelo sistema de controlo, que ao chegar ao servidor é processada e armazenada na base de dados. Esta arquitetura permite também que a plataforma seja apenas utilizada como um repositório de dados, uma vez que a **API REST** possibilita a consulta dos dados guardados. Desta forma o cliente pode utilizar a plataforma para alimentar outros serviços.

### 3.2.1 Requisitos não funcionais

Os requisitos não funcionais, ao contrário dos funcionais, são aqueles que na maior parte das vezes passam despercebidos aos utilizadores da plataforma; contudo são tão ou mais importantes que os funcionais e acabam muitas vezes por gerar restrições ao nível das funcionalidades que podem ser implementadas. No desenho desta plataforma há que ter em conta muitos aspetos não funcionais, de destacar os seguintes:

- Performance;
- Acessibilidade;
- Flexibilidade;
- Usabilidade;
- Segurança;
- Escalabilidade;
- Disponibilidade;
- Portabilidade;
- Eficiência;

De certa forma todos os requisitos anteriores se encontram relacionados, pois todos servem para que as funcionalidades da plataforma sejam possíveis. Sendo esta aplicação uma plataforma de gestão e monitorização, existe uma grande necessidade de disponibilidade, o que obriga a que a plataforma esteja acessível 365 dias por ano. Além disso é preciso garantir que não existem falhas de ligação e que existe redundância de servidores e dados. A quantidade de ligações concorrenciais que a plataforma tem que suportar obrigam a que a plataforma seja bastante eficiente na gestão dos meios disponíveis. É também importante que de uma forma fácil se possa aumentar a capacidade da plataforma tanto em termos de performance como de novas funcionalidades; isto obriga a uma arquitetura de construção que seja flexível ao ponto de ser escalável para se adaptar a novas situações que o mercado possa exigir. Com a quantidade de dados gerados pelos vários sistemas é importante garantir um grande nível de segurança da plataforma para evitar ataques e perdas de dados.

Com a vulgarização da utilização dos smartphones é muito importante garantir que a plataforma se adapta e funciona neste tipo de dispositivos. Este aspeto é de grande importância pois possibilita que os utilizadores possam monitorizar o seu sistema através do smartphone em qualquer lado, desde que possuam acesso à Internet. A plataforma deve assentar num design elegante e bastante fácil de utilizar, deve apresentar-se intuitiva ao utilizador e permitir um acesso rápido a todas as funcionalidades disponíveis.

### 3.2.2 Principais funcionalidades

Depois da breve descrição feita no secção 3.1 , irei agora fazer uma descrição mais detalhada das principais funcionalidades da plataforma e do seu funcionamento lógico. Esta descrição está dividida em 4 partes: a forma como a ligação aos sistemas de controlo é configurada, monitorização, área de utilizador e algumas funcionalidades gerais. As funcionalidades são descritas detalhadamente e explicado também o modo como o utilizador as pode utilizar.

#### Criação dos sistemas

Sendo esta uma plataforma genérica de gestão de sistemas hidropónicos, tem a necessidade de se adaptar a qualquer sistema de controlo, dos mais simples (como os sistemas de controlo *Do It Yourself* (DIY) feitos pelos utilizadores amadores) aos mais complexos. A plataforma será capaz de se adaptar a qualquer tipo de sistema, pois possibilita que o utilizador configure o sistema na plataforma da forma mais adequada ao seu próprio sistema de controlo. Para isso é disponibilizada uma ferramenta de criação de sistemas onde é possível definir todos esses aspetos.

Inicialmente o utilizador terá de criar o seu sistema, onde terá um formulário com a informação básica do sistema (nome, método, tipos de cultura, localização, tamanho). Nesta fase, o utilizador define também se os dados da plataforma podem ser públicos ou privados; esta escolha limita as funcionalidades que o utilizador tem à sua disposição. Se o utilizador escolher manter os dados privados terá várias limitações ao nível das quantidades de dados que pode ter na plataforma, entre outras. De salientar que todos os dados recolhidos na plataforma podem ser utilizados pela mesma para fins de estudos e outros que possam beneficiar o crescimento de conhecimentos sobre hidroponia. Neste caso, a plataforma é obrigada a informar os utilizadores da sua utilização.

Todas as configurações feitas na criação dos sistemas têm funcionalidades *Create, Read, Update, Delete* (CRUD). No final deste passo é dada ao utilizador a chave única que é utilizada nos Header HTTP dos pedidos do sistema. Esta chave serve para identificar e autenticar o sistema na plataforma. Caso não esteja correta ou não exista, os dados não serão aceites pela plataforma.

Na configuração dos sistemas, os utilizadores têm a possibilidade de criar à sua vontade 4 tipos de parâmetros de configuração:

#### VARIÁVEIS

As variáveis são, tal como o nome indica, a ligação entre os dados que o sistema de controlo envia e o que a plataforma recebe. Aqui o utilizador define todas as variáveis que está a enviar para a plataforma e o nome com que vêm no pedido. Pode ainda definir o tipo (inteiro, flutuante, data/hora, booleano, imagem) e a unidade de medida. Este é o passo mais importante na criação do sistema, pois dele

dependem os dados que são gravados na plataforma. Estas variáveis, para além do papel na ligação entre o sistema de controlo e a plataforma, vão servir para outras funcionalidades da plataforma, já que são o que o utilizador pretende monitorizar.

### ALIMENTADORES

Os alimentadores permitem que a plataforma possa recolher dados de outros locais em períodos de tempo definidos. Para isso o utilizador indica o formato em que os dados são recolhidos (**JSON**, **XML**), quais as variáveis que vêm na resposta e que são para guardar, e o URL que a plataforma tem que aceder para recolher os dados. Esta funcionalidade permite, por exemplo, recolher dados de API de previsões de tempo. Os alimentadores servem, tal como as variáveis, para utilizar na monitorização dos sistemas e permitem acrescentar informação aos dados recolhidos pelo sistema de controlo.

### TRIGGERS

Para uma monitorização eficiente e sem preocupações, o utilizador pode configurar triggers com as variáveis e alimentadores. Assim, o utilizador pode criar níveis (nos quais pode escolher a variável e o valor ou intervalo de valores) e qual o alerta que pretende receber. No que diz respeito aos alertas, o utilizador pode definir o texto e a forma como é enviado (Email ou SMS, sendo que o serviço de SMS é pago).

### EVENTOS

Os eventos funcionam de uma forma equivalente aos triggers, mas podem ser configurados para ocorrer em intervalos de tempo e com uma frequência escolhida. Além dos alertas iguais aos dos triggers, o utilizador pode também chamar um URL definido. Nos eventos podem ser definidos backups (**JSON**, **XML**, **CSV**) de dados a enviar por email na periodicidade escolhida. Também podem ser definidos eventos de armazenamento, passando os dados mais velhos (a um tempo definido) para histórico, de forma a não aparecerem no ambiente de monitorização.

Depois de percorridos todos estes passos de configuração da plataforma, é importante disponibilizar ao utilizador alguma documentação e ajuda de como configurar o seu sistema de controlo para enviar os dados para a plataforma. A plataforma tem disponível uma área com a documentação detalhada da API de ligação ao servidor, onde também disponibiliza exemplos de código para Arduino e Raspberry Pi (python). O utilizador é livre de enviar para a plataforma dados de vários sistemas de controlo para o mesmo sistema na plataforma; para isso apenas tem de utilizar a mesma chave única e criar todas as variáveis que os vários sistemas vão enviar. A plataforma disponibiliza, na API de ligação, métodos de leitura de dados. Assim os utilizadores podem utilizar a plataforma como repositório de dados e utilizar a API para alimentar outros serviços. Os dados são

disponibilizados pela API em dois formatos, **JSON** e **XML**.

Fora da criação e configuração dos sistemas, o utilizador tem a possibilidade de introduzir mais alguns dados manualmente. Nesta área é possível introduzir colheitas, quantidades de alimentos recolhidas no dia, assim como perdas de produção e o motivo. Com estes dados, o utilizador consegue ter uma ideia da produção do seu sistema e tirar conclusões sobre o seu maneiio. O utilizador pode também publicar no seu perfil as suas colheitas e deste modo partilhar com outros utilizadores as experiências com o sistema.

### Monitorização

A monitorização está dividida em duas partes: uma é a ferramenta de criação de dashboards, sendo a outra a monitorização com a utilização dos dashboards criados. Nesta área da plataforma, o utilizador pode construir os seus dashboards, onde pode parametrizar todos os aspetos de forma a adaptá-lo à sua visão do sistema e tendo em conta os aspetos que pretende monitorizar. A ferramenta de desenho assenta sobre vários princípios de desenho de dashboards e permite criar (a partir dos dados do utilizador) um número infinito de configurações diferentes de dashboards. O utilizador pode criar a quantidade que desejar de dashboards diferentes. Todo o conteúdo dos dashboards pode ser parametrizável pelo utilizador. A ferramenta disponibiliza as seguintes funcionalidades:

- Vários tipos de gráficos (barras, linhas);
- Barras de progresso;
- Mostradores para valores atuais;
- Adição de texto e imagem.

Todos os elementos anteriores têm um menu próprio de configuração; estes elementos são arrastáveis dentro da área útil de monitorização, que é configurada pelo utilizador. Na criação do dashboard, o utilizador pode escolher o tamanho do ecrã de monitorização e a cor de fundo. Para os gráficos, o utilizador pode escolher quais as variáveis ou alimentadores que pretende ver no gráfico, intervalo e frequência de valores, cores das séries de dados, nome do gráfico, tamanho da letra, cor de fundo. De uma forma equivalente, o utilizador pode configurar os outros elementos que vai adicionando ao seu dashboard e assim criar um dashboard que melhor se adapta à sua situação. Além destes elementos, o utilizador pode ainda adicionar uma área onde é possível visualizar o histórico dos alertas gerados pelos trigger e eventos criados.

Em anexo é possível ver imagens dos testes da construção de um dashboard.

Depois de criar e guardar o seu dashboard, o utilizador pode iniciar a monitorização, que ocorre em tempo real, atualizando os dados em curtos espaços de tempo. Se o utilizador tiver disponível mais do que um sistema e/ou dashboards, ele é livre de navegar entre os vários sistemas e dashboards com os botões de navegação ou teclado.

Sempre que um trigger ou evento gera um alerta, o utilizador além do email ou SMS, é também alertado por um aviso flutuante que aparece no canto superior direito da plataforma, quer o utilizador esteja a monitorizar o seu sistema ou quer esteja simplesmente a navegar na rede social.

### **Área de utilizador**

Tal como em qualquer plataforma, o utilizador terá que se registar e aceitar as regras de utilização; só depois poderá usufruir das funcionalidades da plataforma. O registo é simples e rápido de forma a não ser demasiado extenso para o utilizador. Para se registar, o utilizador apenas terá de preencher nome, email, password e nacionalidade. Os restantes dados podem ser preenchidos a posteriori, pois serão utilizados na rede social em conjunto com os dados do sistema, de forma a originar um perfil de utilizador o mais completo possível.

Na rede social o utilizador pode publicar informação sobre o seu sistema, experiências e artigos. Além disso pode receber comentários e sugestões de amigos que façam ou não parte da sua rede de contactos (depende das configurações de privacidade que o utilizador configura). O utilizador tem sempre disponível na plataforma um chat que pode utilizar para conversar com os seus amigos e com o suporte da plataforma. Além dos amigos, é possível criar grupos privados e partilhar os dashboards com outros utilizadores para que eles possam também monitorizar o seu sistema, sendo esta uma funcionalidade bastante útil para empresas que podem disponibilizar os dashboards para os seus funcionários.

### **Gerais**

Esta plataforma será construída com suporte para multilíngua, inicialmente com 3 línguas: espanhol, inglês e português. Com o crescimento da plataforma e aumento do número de utilizadores nativos de outras línguas a registarem-se, ficarão disponíveis novas traduções da plataforma. Com a recolha dos dados dos sistemas, do perfil de utilizador e utilização da rede social, a plataforma será capaz de criar sugestões de publicações e publicidade dirigida a cada utilizador; essa informação é enviada em emails e apresentada em áreas específicas da plataforma. Este será um dos principais sustento da plataforma, para além da publicidade. A plataforma terá também alguns serviços pagos (ex: SMS, abrir limitações para sistemas privados com vários níveis de abertura).

### 3.3 Design

A interface concentra-se na relação Pessoa-Maquina. Esta é um meio de comunicação para a interação Utilizador-Sistema e uma ferramenta que fornece os instrumentos para esse processo comunicativo. Uma interface deve ser transparente, invisível, sintética, intuitiva e prática.

No desenvolvimento da interface é seguido um design centrado no utilizador, que tem por objetivo tornar a interação deste numa tarefa simples e de forma eficiente em termos de realização dos seus objetivos.

O design implementado envolve os seguintes princípios:

- Proximidade e Alinhamento - elementos que pertençam a um mesmo grupo devem estar próximos, para dar clareza do que está a ser apresentado;
- Equilíbrio, Proporção e Simetria - todos os componentes devem ser complementares;
- Contraste - diferenciar elementos e criar peso em determinados elementos, criar destaque;
- Ordem, Consistência e Repetição - criar regras e segui-las, ou seja, não utilizar layouts diferentes;
- Simplificação - simples, claro e objetivo;
- Legibilidade - os textos existem para serem lidos, ou seja, os textos devem ser apresentados de forma a que o utilizador possa facilmente lê-los;
- Invisível - o design não deve chamar a atenção, tendo por função auxiliar a interação;
- Feedback - retornar ao utilizador informações sobre as ações que foram efetuadas e quais os resultados obtidos;
- Navegação - ser simples, por exemplo, sinalização clara onde o utilizador se encontra;

### 3.4 Usabilidade

É um dos pontos chaves na construção de uma interface, por isso foi-lhe dada muita importância. Para ser possível utilizar a plataforma em tablets e smartphones, a construção da plataforma deve ser *responsive*, adaptando-se automaticamente ao tamanho e orientação dos monitores. Se o utilizador estiver num tablet e o virar de vertical para horizontal, o aspeto visual da plataforma modifica automaticamente, pois tem mais área útil horizontal do que vertical. Este é certamente um aspeto técnico importante, pois permite que a plataforma seja utilizada em qualquer dispositivo e poupa tempo de implementação em aplicações nativas.

Foram já feitos alguns estudos de interface apresentados no anexo A. Este estudo serve de protótipo de baixa fidelidade da plataforma, uma vez que para além dos aspetos de design serviu também para rever as funcionalidades e ergonomia da plataforma.

## 3.5 Testes

### Servidor

De forma a testar o conceito da plataforma foram desenvolvidas algumas funcionalidades (anexo B) a funcionar num servidor local em ambiente de teste (), a esse servidor foram ligados dois exemplos de possíveis sistemas hidropónicos que estavam a recolher dados de humidade e temperatura e a enviar para o servidor. No desenvolvimento do servidor foi utilizado NodeJS, assim como o recurso ao MongoLab para alojar os dados em MongoDB e uma API para o envio de *Short Message Service* (SMS), de destacar as seguintes funcionalidades:

- Configurar e parametrizar sistemas;
- Parametrizar alertas por SMS;
- Receção de dados em tempo real;
- Envio de alertas por sms;

Na plataforma de testes o utilizador pode configurar quais os dados que vão chegar do sistema e parametrizar os nomes que vem no pedido, desta forma o utilizador é livre de adicionar quantos parâmetros quiser e sem limitações por sistema. Este tipo de configuração permite que o utilizador possa abastecer um sistema de um ou mais sistemas, tendo apenas de enviar os diferentes dados para o URL do sistema que foi configurado, isto possibilita que ter vários dispositivos de recolha de informação em locais diferentes e juntar os dados num só local na plataforma. Durante a parametrização o utilizador pode configurar valores de alerta, os quais, se forem ultrapassados, é enviada uma SMS.

Depois de configurado o sistema o utilizador é capaz de monitorizar o seu sistema no dashboard, onde é apresentado um gráfico com a evolução dos valores no tempo e ainda mostradores rápidos com o valor atual, máximo e mínimo.

### Sistemas

Para os testes, foram utilizados dois arduinos (duemilanove e mega 2560), ambos com o sensor de temperatura e humidade DHT11<sup>1</sup>, e um deles com mais o sensor de temperatura DS18B20<sup>2</sup>. Para possibilitar enviar os dados, uma vez que não possuem placa de rede, a

<sup>1</sup><http://www.micropik.com/PDF/dht11.pdf>

<sup>2</sup><http://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>

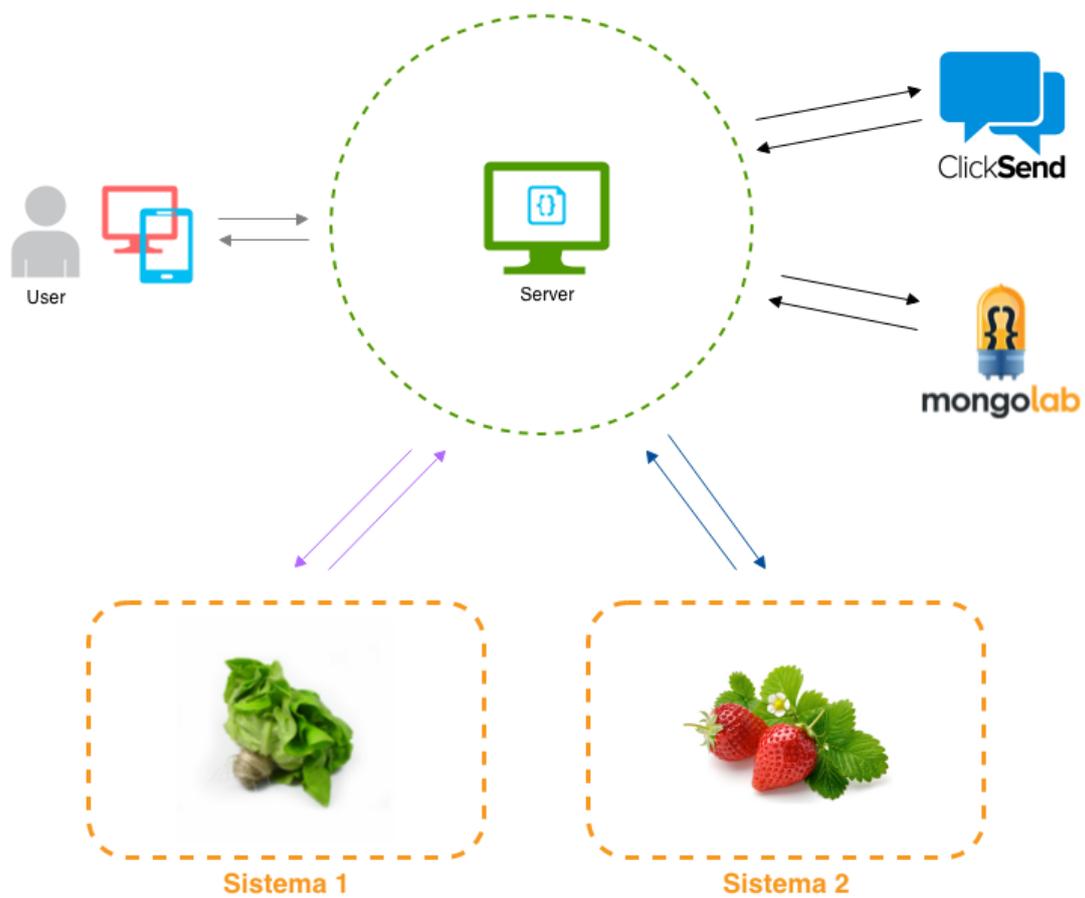


Figura 3.3: Diagrama do sistema de testes

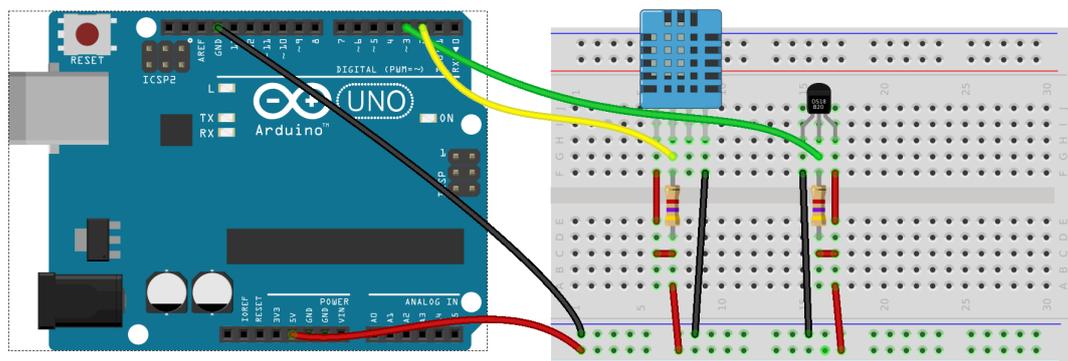


Figura 3.4: Configuração do arduino

solução foi ligar os arduinos por porta série a um Raspberry Pi, no qual foi desenvolvido um script python que comunicava com o arduino, tratava os dados e fazia o pedido ao servidor. Os dados são enviados do arduino numa string para o python, o qual é responsável por fazer o parsing do texto e seguidamente fazer o pedido POST ao servidor, é neste ponto que é utilizada a configuração que foi feita no servidor, no pedido POST os dados enviados tem como nome as configurações feitas. No anexo C podemos ver a simplicidade do código de um sistema e como é fácil enviar por um sistema a funcionar.

Com este sistema foi possível recolher dados de dois sistemas em simultâneo e serve como objeto de estudo para futura implementação. Estes teste teriam agora de passar de um ambiente local para um ambiente mais abrangente de forma a recolher informações sobre a melhor forma de otimizar tanto o sistema como a API de recolha de dados, esses dados seriam maioritariamente recolhidos através de opiniões de utilizadores de teste reais.



# Capítulo 4

## Conclusões

### 4.1 Conclusão

Durante toda a pesquisa efetuada no decorrer da tese, concluí que a hidroponia é uma técnica difícil de dominar, exigente e que necessita de uma constante monitorização. Estes aspetos, apesar das inúmeras vantagens que a hidroponia apresenta, afastam muitas pessoas de adotarem esta técnica para as suas culturas. A informação existente sobre hidroponia encontra-se dispersa pela Internet, é possível consultar bastantes artigos e informação fidedigna no site da FAO (<http://www.fao.org>) . Uma plataforma como a apresentada nesta tese iria concentrar várias fontes de informação, dados recolhidos pelos sistemas ligados à plataforma e informação colocada pelos utilizadores nos seus perfis num só local. Esta informação será certamente útil para se realizarem novos estudos sobre o tema.

A escolha de tecnologias recai em tecnologias web extremamente dinâmicas, isto porque sendo os sistemas hidropónicos dinâmicos (variações de temperatura, pH, nutrientes, humidade e luz), é necessário que a plataforma vá ao encontro das necessidades dos utilizadores, permitindo com que facilmente tenham acesso ao estado atual dos seus sistemas, e assim conseguirem entender a sua evolução. Este dinamismo da plataforma facilita o manejo hidropónico e possibilita que qualquer pessoa possa experimentar ter o seu próprio sistema. A tecnologia de ligação dos sistemas de controlo é open-source, pois deste modo, além da informação que é disponibilizada na plataforma, os utilizadores têm acesso a mais informação em comunidades espalhadas pela Internet.

## 4.2 Trabalho futuro

Futuramente seria interessante juntar a tecnologia de Websockets à API de ligação. Os Websockets funcionam em cima do protocolo TCP-IP, mas ao contrário das ligações normais (que apenas funcionam num sentido), os Websockets mantêm a ligação aberta. Permitem ainda que a informação tenha origem dos dois lados; assim os utilizadores poderiam receber atualizações sem a necessidade de pedidos constantes para atualizar as páginas. Outro trabalho que poderia tornar a plataforma ainda mais competitiva, seria a possibilidade de controlar os seus sistemas, através da implementação de algoritmos de controlo do lado da plataforma, que atuaria sobre o sistema de controlo.

Com o aumento do número de dados e de informação gerada pelos utilizadores, podem ser desenvolvidos algoritmos que, com base nos dados recolhidos nos sistemas, aconselhem formas de melhorar os seus sistemas aos utilizadores.

O passo seguinte seria passar a plataforma para testes com um numero maior de utilizadores e ao mesmo tempo continuar a implementar as funcionalidades em falta, de forma a poder conduzir o desenvolvimento de acordo com os inputs dos utilizadores.

# Bibliografia

- [1] Estar presente na web não basta... a usabilidade aumenta a eficiência das empresas. Last Access: 01-09-2014.
- [2] Mysql. <http://www.mysql.com>. Last Access: 01-09-2014.
- [3] Trygve m. h. reenskaug mvc-xerox parc 1978-79. <http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html>. Last Access: 01-09-2014.
- [4] fontes das imagens de aeroponia. <http://www.organicgardensupply.com/wp-content/uploads/2013/08/aquaponics.jpg> - <http://www.hydroponicist.com/pages/images/bubbler-dwc-system.jpg>, 2013. Last Access: 01-09-2014.
- [5] fontes das imagens de aeroponia. <http://aerofarms.com/wordpress/wp-content/themes/aerofarms/images/why/smarter/technical-diagram.jpg> - <http://boards.cannabis.com/attachments/hydroponics/43503d1135826890-help-aerponic-clogged-misters-spray.jpg> - <http://revistagloborural.globo.com/Revista/GloboRural/foto/0,,43968735,00.jpg>, 2013. Last Access: 01-09-2014.
- [6] *Hydroponic Food Production: A Definitive Guidebook for the Advanced Home Gardener and the Commercial Hydroponic Grower, Seventh Edition*. CRC Press, 7 edition, 2013.
- [7] ALLAN. How to use model-view-controller (mvc). <http://heim.ifi.uio.no/trygver/themes/mvc/mvc-index.html>. Last Access: 01-09-2014.
- [8] Gary Anthes. Html5 leads a web revolution. *Commun. ACM*, 55(7):16–17, July 2012.
- [9] Arduino. Arduino. <http://www.arduino.cc>. Last Access: 01-09-2014.
- [10] Olga Baysal, Reid Holmes, and Michael W. Godfrey. Developer dashboards: The need for qualitative analytics. *IEEE Software*, 30(4):46–52, 2013.
- [11] Daniel Chudnov. Restful http. *Computers in Libraries*, 27(4):20, Abril 2007.

- [12] DIMAP. Design de interface de utilizador. <http://www.dimap.ufrn.br/~jair/ES/c6.html>. Last Access: 01-09-2014.
- [13] exceldashboardtemplates. exceldashboardtemplates. <http://www.exceldashboardtemplates.com/how-to-add-a-line-to-an-excel-chart-data-table-and-not-t>. Last Access: 01-09-2014.
- [14] FAO. Soilless culture information system. <http://www.fao.org/hortivar/scis/scis.htm?TRX=Redirect&T0=IDX>, 2013. Last Access: 01-11-2014.
- [15] FAO. Food for the cities. 2014. <ftp://ftp.fao.org/docrep/fao/012/ak824e/ak824e00.pdf>.
- [16] FAO. *A Vegetable Garden for All*. FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS, 2014. <http://www.fao.org/docrep/019/i3556e/i3556e.pdf>.
- [17] Vandemar Faquin and Pedro Roberto Furlani. Cultivo de hortaliças de folhas em hidroponia em ambiente protegido. *Informe Agropecuário*, 1999.
- [18] Stephen Few. *Information Dashboard Design: The Effective Visual Communication Of Data*. O'Reilly Media, Inc, USA, feb 2006.
- [19] FOOD and AGRICULTURE ORGANIZATION OF THE UNITED NATIONS. Urban and peri-urban agriculture. *SPECIAL PROGRAMME FOR FOOD SECURITY*, page 84, Julho 2001. [http://www.fao.org/fileadmin/templates/FCIT/PDF/briefing\\_guide.pdf](http://www.fao.org/fileadmin/templates/FCIT/PDF/briefing_guide.pdf).
- [20] FOOD and AGRICULTURE ORGANIZATION OF THE UNITED NATIONS. Growing greener cities. page 19, 2010. <http://www.fao.org/ag/agp/greenercities/pdf/ggc-en.pdf>.
- [21] Marcelo Gluz. Principios e dogmas de design para a internet. <http://webinsider.uol.com.br/2008/04/04/principios-e-dogmas-de-design-para-a-internet>. Last Access: 01-09-2014.
- [22] Google. Angularjs. <https://angularjs.org/>. Last Access: 01-09-2014.
- [23] Keith Griffin and Colin Flanagan. Defining a call control interface for browser-based integrations using representational state transfer. *Comput. Commun.*, 34(2):140–149, February 2011.
- [24] César Marulanda and Juan Izquierdo. *Popular hydroponics gardens*. 1993. [http://www.fao.org/alc/file/media/pubs/2000/popular\\_hydroponic\\_gardens.pdf](http://www.fao.org/alc/file/media/pubs/2000/popular_hydroponic_gardens.pdf).
- [25] menkhit. Html5 + css3 + javascript. <http://www.menkhit.com/conception-web.html>. Last Access: 01-09-2014.

- [26] Papadopoulos I. Polycarpou P., Chimonidou D. Techniques for improving water use efficiency in greenhouse cultivation in cyprus. *Options Méditerranéennes : Série B. Etudes et Recherches; n. 57*, 2007. <http://om.ciheam.org/om/pdf/b57/00800779.pdf>.
- [27] Luli Radfaher. Principios de design. <http://www.luli.com.br/2008/03/20/3-principios-de-design>. Last Access: 01-09-2014.
- [28] Roberto Rezende, Celso Helbel Junior, Paulo Sérgio Lourenço de Freitas, Antonio Carlos Andrade Gonçalves, Rivanildo Dallacort, and José Antônio Frizzone. Diferentes soluções nutritivas aplicadas em duas vazões na produção hidropônica da cultura da alface. *IRRIGA*, 12(3):354–363, Julho-Setembro 2007.
- [29] Craig Schiff. Three things you should know about dashboards. *DM Review*, 18(6):29, Junho 2008.
- [30] sdhydroponics. Sistema gota-gota. [http://sdhydroponics.com/resources/wp-content/uploads/2012/05/tomato\\_rockwool.jpeg](http://sdhydroponics.com/resources/wp-content/uploads/2012/05/tomato_rockwool.jpeg), 2013. Last Access: 01-09-2014.
- [31] T. B. Sheridan. Adaptive automation, level of automation, allocation authority, supervisory control, and adaptive control: Distinctions and modes of adaptation. *Trans. Sys. Man Cyber. Part A*, 41(4):662–667, July 2011.
- [32] Zachary Thacher. Responsive mobile site design. *GCI*, page 34, Abril 2014.
- [33] w3c. html5. [www.w3.org/TR/html5/](http://www.w3.org/TR/html5/). Last Access: 25-08-2014.
- [34] w3c. Html5 diferencences. [www.w3.org/TR/2012/WD-html5-diff-20120329](http://www.w3.org/TR/2012/WD-html5-diff-20120329). Last Access: 31-08-2014.
- [35] w3schools. Css3. <http://www.w3schools.com/css3/default.asp>. Last Access: 01-09-2014.
- [36] w3schools. Html5. <http://www.w3schools.com/html5/default.asp>. Last Access: 01-09-2014.
- [37] w3schools. Javascript. <http://www.w3schools.com/js/default.asp>. Last Access: 01-09-2014.



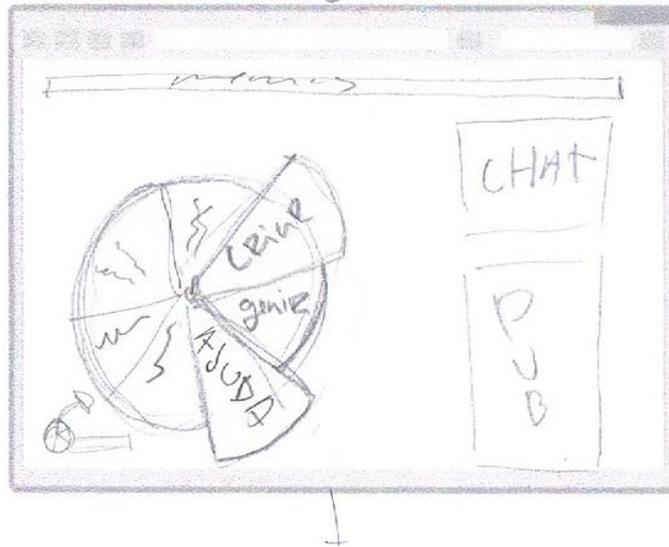
# Anexos



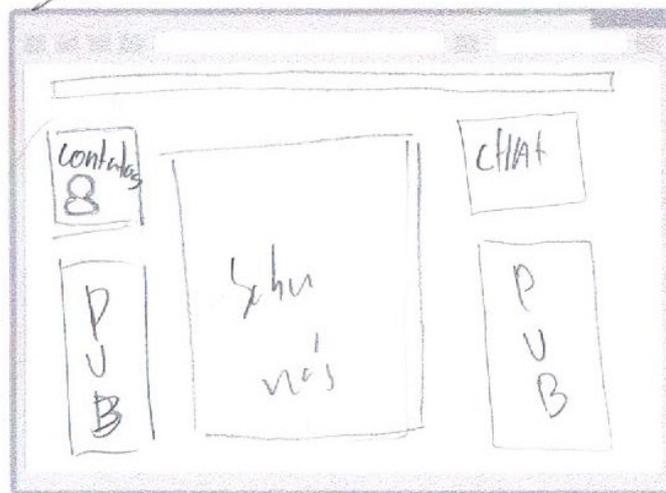
Anexo A

Estudios de interface

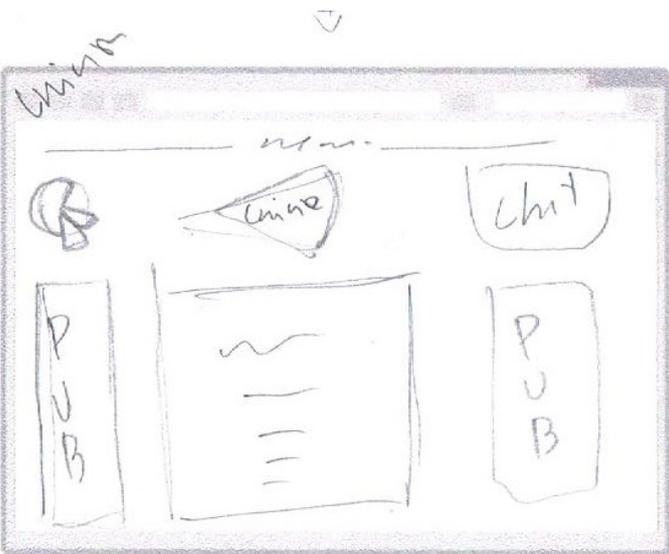
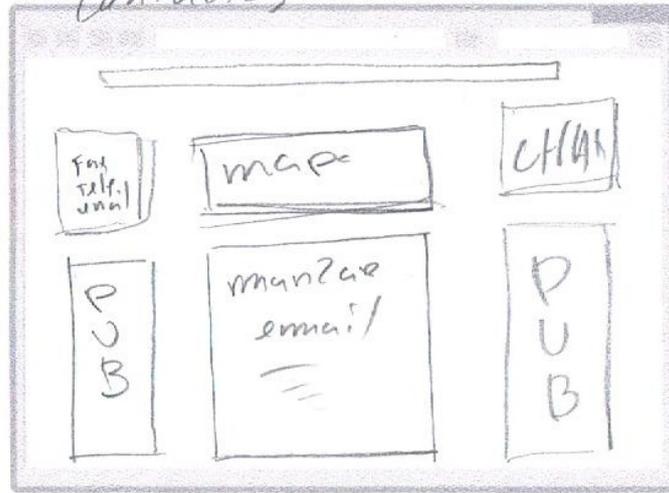
ChinP  
Desn BOPZ

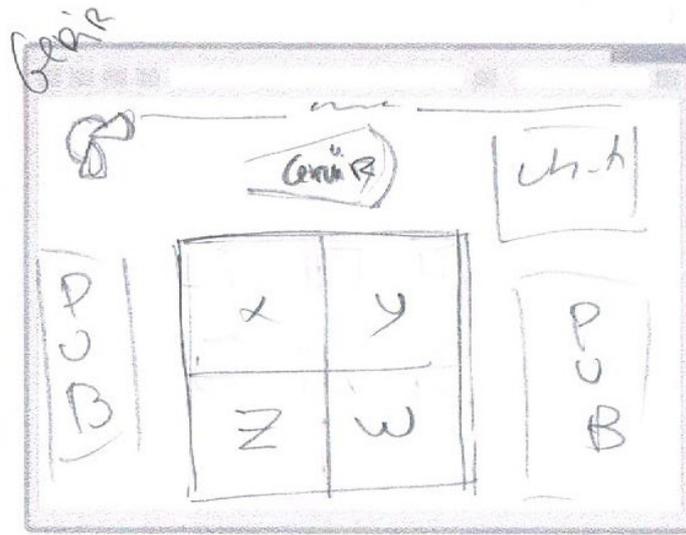


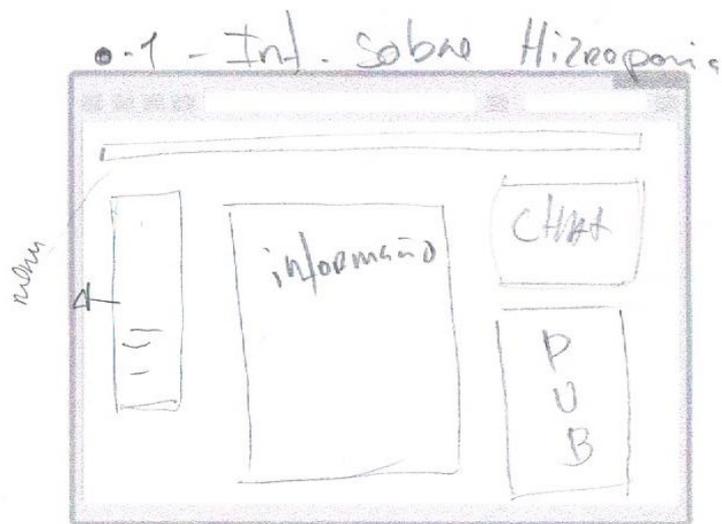
conhe nós



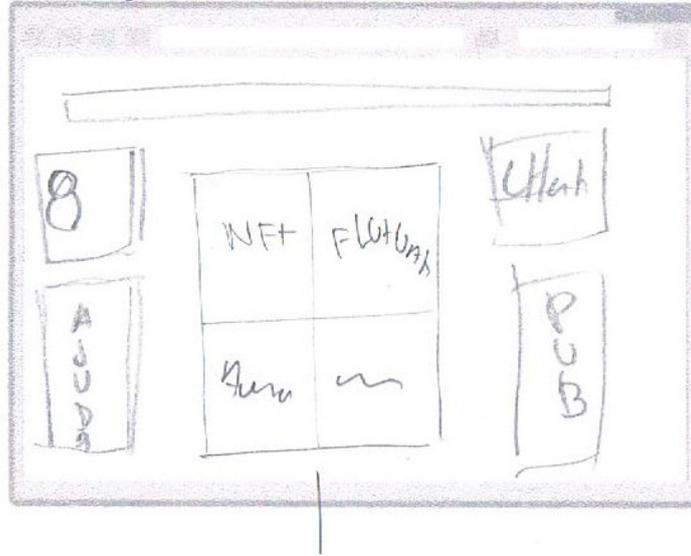
contactos





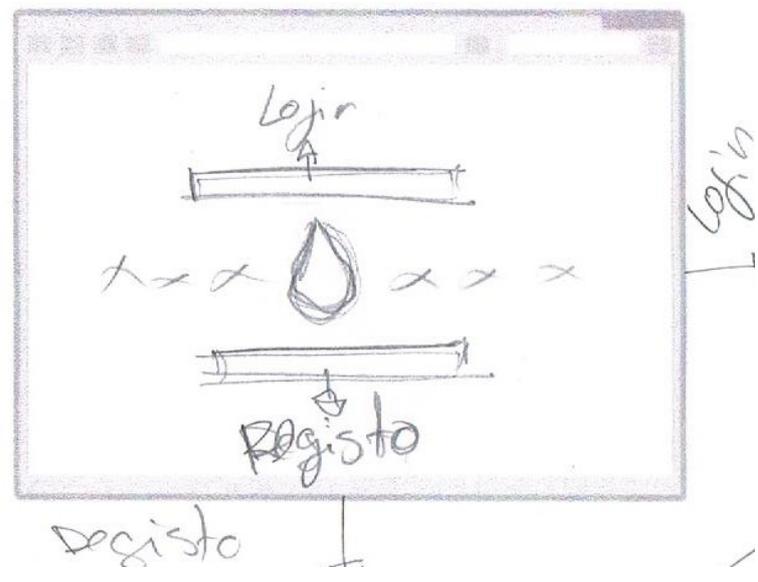
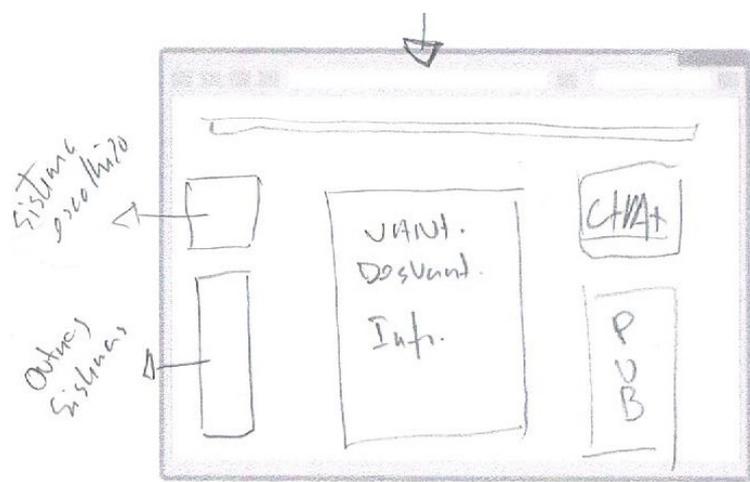


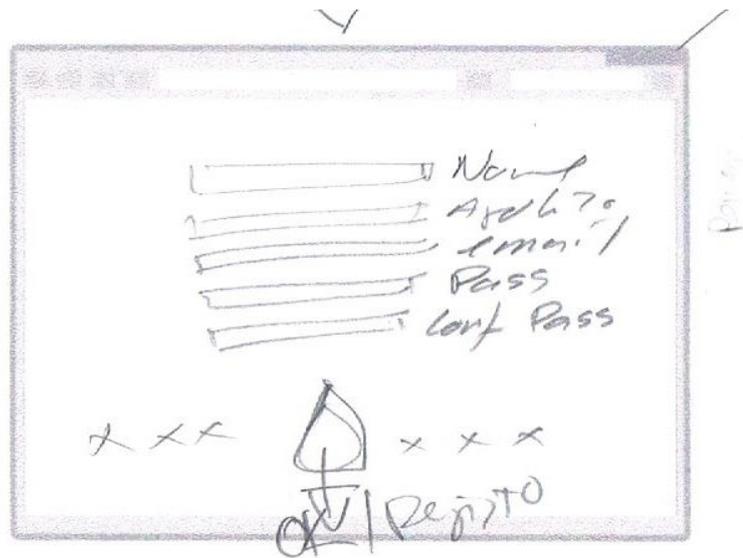
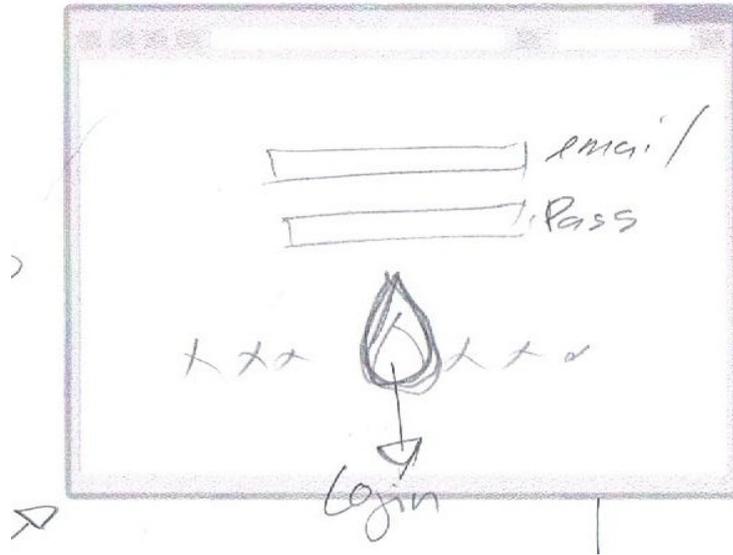
Sistemas Híbridos

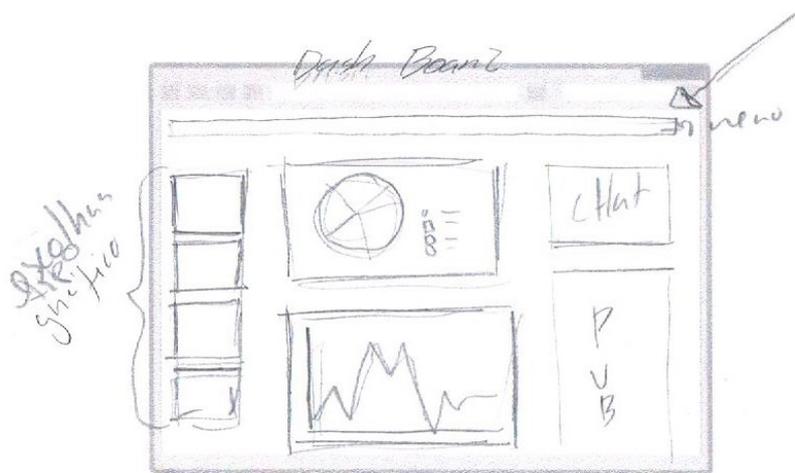


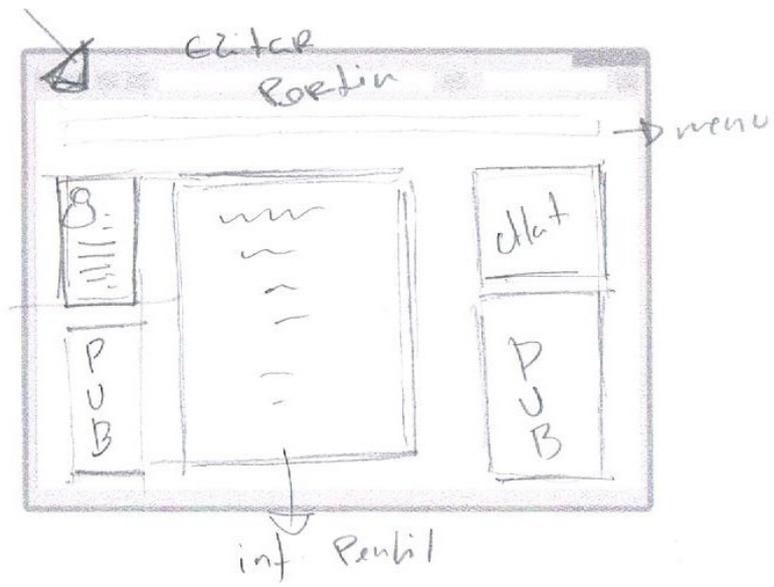
Recolha de dados (live)











**Anexo B**

**Plataforma testes**

MYHYDRO NETWORK MY SYSTEMS MY PAGE DOCUMENTATION Martins

Everything you need is here  
Access data, people, and discover systems around you.

**Last Updates**

- PH (Strawberries): 6.50
- Temperature (Strawberries): 29.38
- Humidity (Strawberries): 78.35

**News**

**Most popular**

- MP hydroponic 1
- MP hydroponic 3
- MP hydroponic 3

**Latest**

- LT hydroponic 1
- LT hydroponic 3
- LT hydroponic 3

**People you may know**

My systems

NEW MY SETTINGS

**STRAWBERRIES**

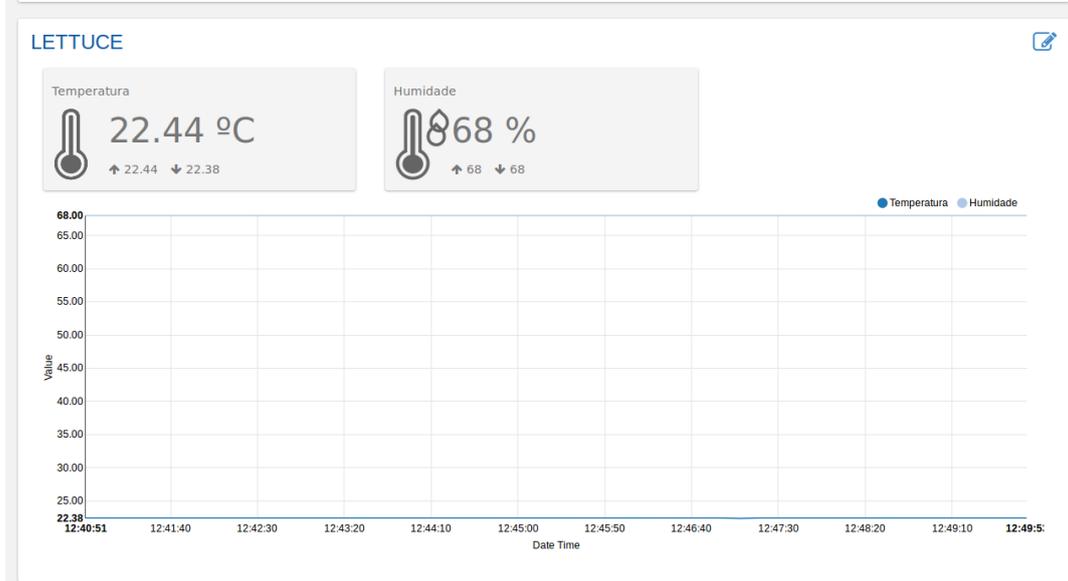
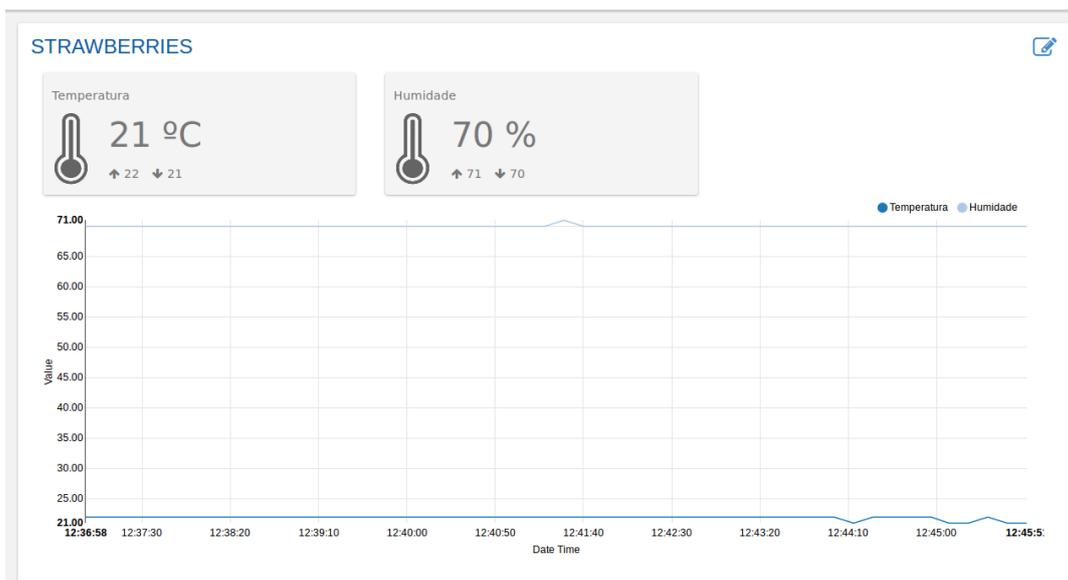
Nome: Strawberries

**Valores**

Name	Temperatura	Key	temp
Max	50	Min	0
Image	/static/img/dashboard/temperatura.png	Simbol text	°C
Name	Humidade	Key	hum
Max	90	Min	0
Image	/static/img/dashboard/temperatura.png	Simbol text	%



NEW MY SETTINGS





# Anexo C

## Código de um sistema

### C.1 Código Arduino do sistema

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include "DHT.h"

#define ONE_WIRE_BUS 3
#define DHTPIN 2
#define DHTTYPE DHT11

OneWire oneWire(ONE_WIRE_BUS);
DHT dht(DHTPIN, DHTTYPE);
DallasTemperature sensors(&oneWire);

void setup() {
  Serial.begin(9600);
  dht.begin();
  sensors.begin();
}

void loop() {
  delay(10000);
  float h = dht.readHumidity();
  sensors.requestTemperatures();
  float t = sensors.getTempCByIndex(0);
  String data = String(h)+"|" +String(t);
```

```

    Serial.println(data);
}

```

## C.2 Código python de interface

```

import time
import serial
import json
import urllib2

lastTemp = 0
lastHum = 0
url = "http://api.<ipServidor>/v1/tesedata/system/Lettuce?apiKey=<key>"
ser = serial.Serial('/dev/ttyACM0', 9600) #Porta do arduino
while True:
    time.sleep(10);
    try:
        if ser.inWaiting():
            try:
                #informa o vem por porta serie no formato
                #<strHumidade>|<strTemperatura>
                data = ser.readline().strip().split("|") #Ler info do arduino
                lastTemp = data[1]
                lastHum = data[0]
            except:
                data = []
                data.append(lastHum)
                data.append(lastTemp)
            pass
        print data
        try:
            dataSend = json.dumps({ "temp" : data[1], \
                                   "hum": data[0], \
                                   "dateTime": int(time.time()) })
            clen = len(dataSend)
            req = urllib2.Request(url, \
                                  dataSend, {'Content-Type': 'application/json', \
                                              'Content-Length': clen})
            f = urllib2.urlopen(req)
            response = f.read()
            print response
            f.close()
        except:

```

```
        pass
except :
    pass
```

