



RegDuino



Javier Molina Suárez
Pla d'estudis: Grau Multimèdia
Area: TFG Arduino
Consultor: Antoni Morell Pérez
Responsible Area: Pere Tuset Peiró
Data: 09/06/2019



Aquesta obra està subjecta a una llicència de
[Reconeixement-CompartirIgual 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-sa/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball	REGDUINO
Nom de l'autor	Javier Molina Suárez
Nom del consultor	Antoni Morell Pérez
Nom del PRA	Pere Tuset Peiró
Data lliurament	06/2019
Titulació o programa	Grau Multimèdia
Àrea de Treball Final:	TFG Arduino
Idioma del treball	Català
Paraules clau	Arduino, reg, automàtic
Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i> Resum del treball	
<p>RegDuino, és un sistema de reg autogestionat, que consisteix a fer que una planta auto reguli l'aigua que necessita per alimentar-se.</p> <p>Aquesta aigua la podem aconseguir del subministrament de casa o d'un bidó que s'ompli amb la pluja d'aigua o de forma manual.</p> <p>Aquest subministrament es farà en funció de diferents valors, que els aconseguirem segons uns sensors que instal·larem a la plantació, per agafar la humitat de la terra, la temperatura, la llum, ... Aquests ens donaran la informació que necessitem per poder anar subministrant l'aigua en funció de si necessita o no necessita la plantació.</p>	
Abstract (in English, 250 words or less):	
<p>RegDuino, is a self-managed irrigation system, which consists in the fact that a self-regulating plant regulates the water it needs to feed itself.</p> <p>This water can be obtained from the supply of a house or a dump that is filled with rain or manually.</p> <p>This supply will be made based on different values, which will be achieved according to the sensors that we will install in the plantation, to take the humidity of the earth, the temperature, the light ... These will give us the information we need to be able to Go to supply the water depending on whether you need or do not need the planting.</p>	

Index

Introducció	5
Objectiu	6
Viabilitat	11
Pla d'inversions inicials.	11
Pla de finançament	12
Previsió de tresoreria i el sistema de cobrament a clients i pagament a proveïdors.	13
La previsió del compte de resultats	13
L'ànàlisi del llinar de rendibilitat o punt d'equilibri	14
L'avaluació del projecte.	14
Planificació	16
Disseny	16
Diagrama de Gantt	17
Pressupost	18
Modalitats	19
Sistema de suministrament d'aigua.	20
Tipus de Reg	22
Gestió de Riscos	25
Caudal d'aigua	28
Plataforma web	29
Plataforma Regduino	29
Diagrama de Conexió	30
Casos d'us	31
Diagrama d'Estats dels sensors	34
Diagrama de flux dels sensors	35
Diagrama de seqüència	36
Pantallas	37
Pantalla principal	37
Pantalla sensors	37
Pantalla registre	38
Pantalla About	38
Plataforma Thinger.io	39
Introducció	39
Funcionament	39
Pantalla del llistat dels dispositius connectats.	41
Pantalla estadístiques del Dispositiu connectat	41
Dashboard	42

Widgets per la interacció amb la placa (dins del dashboard)	42
Firmware per Arduino Yun	43
Sistema arduino	44
Arduino YUN.	44
Descripció	44
Disseny connexions	44
Codifont	44
Sensor humitat	47
Descripció	47
Disseny connexió	47
Codifont	48
Sensor temperatura	49
Descripció	49
Disseny connexió	49
CodiFont	50
Sensor Llum	51
Descripció	51
Disseny connexió	51
Codifont	52
Electro Valvula	53
Descripció	53
Dades tècniques	53
Disseny connexió	53
Codi font	54
Enviar dades al servidor web	55
Impediments del projecte	57
Annex 1	58
Referencias	59

Introducció

RegDuino, és un sistema de reg autogestionat, que consisteix a fer que una planta auto reguli l'aigua que necessita per alimentar-se.

Aquesta aigua la podem aconseguir del subministrament de casa o d'un bidó que s'ompli amb la pluja d'aigua o de forma manual.

Aquest subministrament es farà en funció de diferents valors, que els aconseguirem segons uns sensors que instal·larem a la plantació, per agafar la humitat de la terra, la temperatura, la llum, ... Aquests ens donaran la informació que necessitem per poder anar subministrant l'aigua en funció de si necessita o no necessita la plantació.

Objectiu

- **Objectius i justificació**

L'objectiu principal és la creació d'un sistema de reg autònom. D'aquesta manera la planta/es tindran aigua segons les necessitats.

- **Descripció**

Consistirà en un sistema Arduino amb uns sensors connectats a la placa i una electrovàlvula que connectarà el sistema de reg amb la sortida d'aigua. Aquest sistema autònom, estarà enviant dades a una web on l'usuari podrà visualitzar l'estat del sistema en aquell moment.

- **Usuaris**

El públic objectiu de l'aplicació és molt ampli, ja que el sistema de reg pot ser independent del sistema de reg. Podem regar des d'una planta, fins a un hort petit. Però principalment, el públic objectiu és per als usuaris que vulguin aprofitar al màxim l'aigua per a les plantes, que es faci de forma autònoma i gastant només el que la planta necessita. Inicialment aquest sistema està enfocat a un lloc de terra petit, però la idea és que aquest sistema evolucioni perquè en un futur pugui ser un sistema per a un camp de tomàquets o un camp de carxofes.

1) Anàlisi de continguts

a) Objectius generals

- Sistema de reg autònom

b) Objectius específics

- Detectar la humitat de la terra:
 - valor sensor sec 1023, 0%
 - valor sensor humit 0. 100%.
 - Es considerara valor < 400 , 50% humitat per obrir la electrovàlvula
- Detectar la temperatura ambient:
 - Es considerara valor ≤ 15 graus per obrir la electrovàlvula
- Detectar si es de dia o de nit
 - valor sensor llum màxim 900, fosco
 - valor sensor llum mínim 0, llum
 - Es considerarà valor ≤ 400 , de dia
 - Es considerarà valor > 400 , de nit

- Visualitzar els valor dels sensors a distancia.
 - regduino.javims.es, els valors s'actualitzaran cada 3 segons
 - thinger.io els valors s'actualitzant segons la placa ho rep.
- Enregistrar els valors dels sensors per poder visualitzar la trajectòria.
 - En el cas dels sensors llum, humitat i temperatura, la placa enviara cada 60 segons les dades a la bbdd de la web regduino.javims.es
 - En el cas de l'electrovàlvula, el valor s'enviarà quan s'obri o es tanqui l'electrovàlvula.
- Obrir i/o tancar l'electrovàlvula.
 - Mode Automàtic, segons les consideracions dels valors exposades anteriorment.
 - Mode Manual, a través de la plataforma thinger.io es podrà manipular amb un botó.

2) Anàlisi formal

a) Estil artístic: Part Visual de la web

- **Estil genèric:** Estil simple, una capçalera, un menú i el contingut que anirà canviant en clicar a les parts del menú.
- **Aspectes artístics:** color verd i blanc, color per excel·lència de la naturalesa.
- **Esbossos o maquetes:**

RegDuino



Disseny web

b) Tipologia dels mitjans

S'utilitzarà text per representar els valor i imatges per representar símbols o elements físics.

Formes d'interacció: descriurem quins tipus d'interacció oferirà l'aplicació o producte:

- **Interacció del ratolí:** clics per el menú de la pàgina per visualitzar la informació
- **Tipus de menús:** menú a la part de dalt de la pàgina, i mitjançant clics es podrà moure per el

c) Navegació

Es una navegació senzilla, té 4 menús que donen informació relativa als sensors de la placa arduino.

d) Estructura de pàgina

Pàgina principal. Descripció del sistema que gestiona.

Menú sensors. Tenim la informació, que envia la placa Arduino dels sensors connectats.

Menú registre. Tenim el registre dels sensors que va enviant la placa en dos moments del dia.

Menú col·laboració, amb la informació de l'autor i col·laboració.

e) Dimensió de l'aplicació

La pàgina té 4 pàgines, i és una web estàtica, ja que no hi ha massa interacció amb l'usuari.

D'altra banda tenim una altra pàgina ja existent, que és thinger.io, amb la que la placa també es comunicarà, per poder interactuar amb l'aixeta per obrir i tancar-la.

I el sistema base dels sensors i la placa Arduino. El firmware de la placa que es comunica amb les webs.

D'una banda tenim un firmware per a thinger.io i d'altra banda tenim un firmware per a la web pròpia.

3) Anàlisi tecnològica

a) Solució tecnològica: Les tecnologies que s'utilitzaran seran les següents.

Web: PHP, html, css, javascript

Bases de dades: SQL, Mysql

Arduino: C++ , librerias

b) Requeriments de producció: Para la producció d'aquest projecte es necessiten els següents components.

Placa Arduino Yun

Sensor temperatura

Sensor humitat terra

Sensor llum

Electrovàlvula

Relay

Galleda o Connectors toma d'aigua

Un test

c) Requeriments d'usuari:

Ordinador per visualitzar les dades i obrir tancar l'aixeta en el cas que sigui necessari.

4) Necessitats de manteniment

Es contempla actualitzacions del producte per millorar les funcionalitats existents i per afegir de noves.

5) Calendaris i terminis

Adjuntat el diagrama de gantt amb la planificació

6) Pressupost

Adjuntat a l'apartat del pressupost.

Viabilitat

Pla d'inversions inicials.

Materials	Unitats	Preu
Electrovàlvula	1	16€
Placa Arduino	1	70€
Sistema de reg	1	20€
Tubs subministrament aigua	1	10€
Ordinador	1	400€
Hosting virtual i Domini	1	20€
Sensor temperatura	1	2€
Sensor Humitat terra	1	2€
Sensor llum	1	2€
Software		
Arduino IDE		0
Navegador web		0
Editor text		0
Filezilla		0
Recursos humans		
Salari personal + formacio	160h	40€/h
Total		6942€

Pla de finançament

El pla de finançament, consisteix en la venda del sistema de reg, el suport i el manteniment d'aquest. Serà de la següent manera:

Material		142€
Alta inicial		45€

Suport/formacio	puntual (15 min)	10€
Manteniment	mensual	20€
Muntatge parcial	50h	2000€
Muntatge complet	160h	6400€

Material - Tot el material físic que es necessita per al muntatge.

Alta inicial - Instal·lació software a la placa Arduino i formació dels sensors

Suport/formació - Consultes relacionades amb el sistema o placa Arduino.

Manteniment - Actualitzacions del sistema de reg, millores o correcció d'errors.

Muntatge parcial - Muntatge, de sensor, placa i electrovàlvula, software.

Muntatge complet - sistema de reg i sistema de subministrament d'aigua.

Client 1 - Material.

Client 1		
Sistema de (reg material)	142€	142€
Alta inicial		45€
Suport (opcional)		10€/puntual
Total		192€

Client 2 - Muntatge Inicial

Client 2		
Sistema de reg	1	142€
Salari personal	50h	40€/h
Alta inicial	1	40€
Suport (opcional)	1	10€/puntual
Total		2192€

Client 3 - Muntatge + manteniment + suport

Client 3		
Sistema de reg	1	142€
Salari personal	160h	40€/h
Manteniment	1	20€/mes
Suport (opcional)	1	10€/puntual
Total		6592€

Previsió de tresoreria i el sistema de cobrament a clients i pagament a proveidors.

Annex 1

La previsió del compte de resultats

Annex 1

L'ànàlisi del líndar de rendibilitat o punt d'equilibri

Per recuperar la inversió inicial, és necessitarà la venda de 1400 unitats d'aquest producte, ja que el benefici per al producte bàsic és de 5 € entres els costos i els ingressos.

Contant que el personal trigui una hora a donar d'alta el sistema. Que segurament es podrà reduir a la meitat. Per tant el benefici serà més alt i amb menys unitats es podrà recuperar la inversió inicial més aviat.

Ingresos			
Concepte	Unitats	€/unidad	Total
Material	1400	182,00 €	254.800,00 €
Alta	1400	45,00 €	63.000,00 €
Ingresos			317.800,00 €
Costos			
Material	1400	182,00 €	254.800,00 €
Personal (h)	1400	40,00 €	56.000,00 €
Costos			310.800,00 €
Beneficio			7.000,00 €

L'avaluació del projecte.

Vist que s'han de vendre 1400 unitats del projecte per a poder arribar a recuperar el capital invertit. Aquesta és la modalitat bàsica, però arran d'aquest producte també es deriven

altres serveis com el muntatge parcial o complet, suport i manteniment. Que poden derivar més beneficis però també més costos.

La idea bàsica d'aquesta modalitat és que tu béns el producte bàsic i et desentens del suport i del manteniment, ja que també s'entrega el firmware de la placa, ja que és codi lliure.

Planificació

Disseny

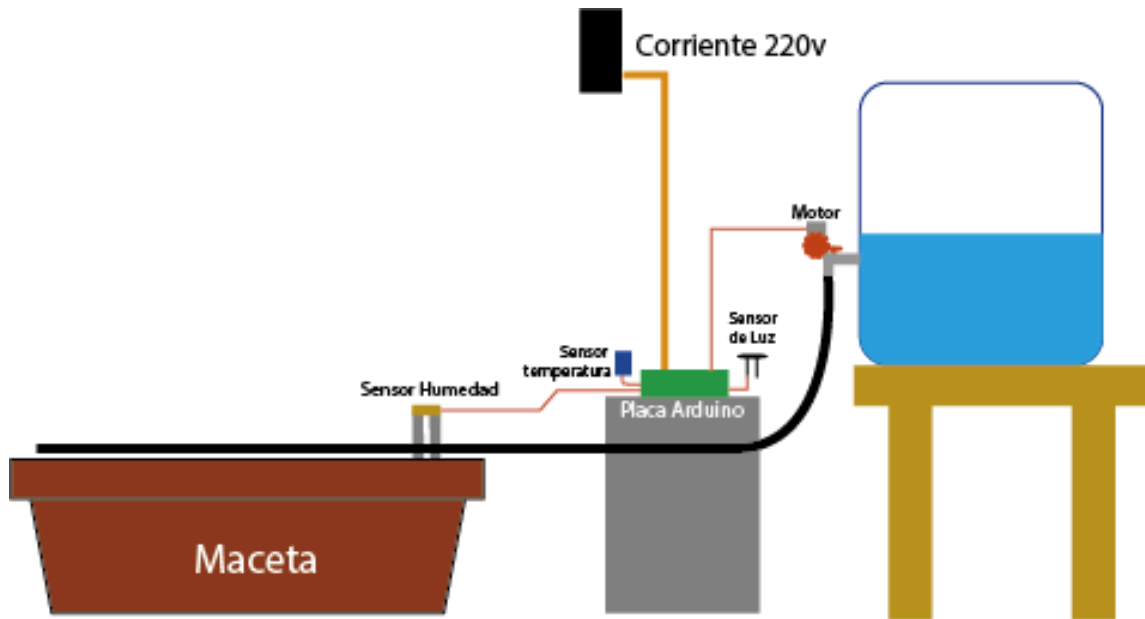
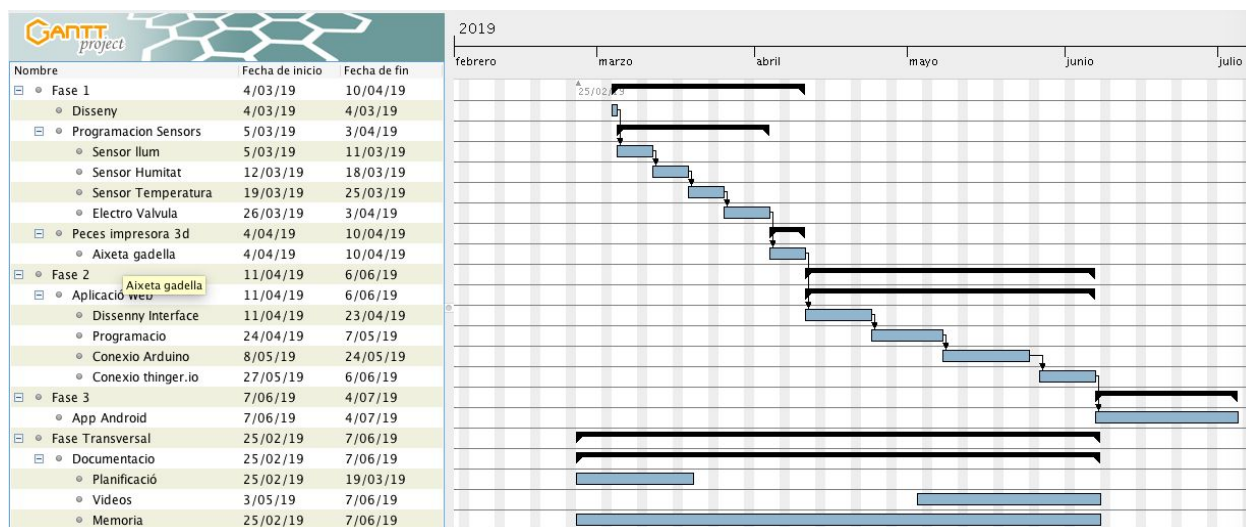


Diagrama de Gantt



Nombre	Fecha de inicio	Fecha de fin
▼ Fase 1	4/03/19	10/04/19
○ Disseny	4/03/19	4/03/19
▼ Programacion Sensors	5/03/19	3/04/19
○ Sensor llum	5/03/19	11/03/19
○ Sensor Humitat	12/03/19	18/03/19
○ Sensor Temperatura	19/03/19	25/03/19
○ Electro Valvula	26/03/19	3/04/19
▼ Peces impresora 3d	4/04/19	10/04/19
○ Aixeta gadella	4/04/19	10/04/19
▼ Fase 2	11/04/19	6/06/19
▼ Aplicació Web	11/04/19	6/06/19
○ Dissenny Interface	11/04/19	23/04/19
○ Programacio	24/04/19	7/05/19
○ Conexio Arduino	8/05/19	24/05/19
○ Conexio thinger.io	27/05/19	6/06/19
▼ Fase 3	7/06/19	4/07/19
○ App Android	7/06/19	4/07/19
▼ Fase Transversal	25/02/19	7/06/19
▼ Documentacio	25/02/19	7/06/19
○ Planificació	25/02/19	19/03/19
○ Videos	3/05/19	7/06/19
○ Memoria	25/02/19	7/06/19

Pressupost

Cost dels materials i dels serveis.

Article	Unitats	Preu/unitat	Total
Electrovalvula	1	16€	16€
Placa Arduino	1	70€	70€
Sistema de reg	1	20€	20€
Tubs suministrament aigua	1	10€	10€
Ordinador	1	400€	400€
Hosting virtual i Domini	1	20€	20€
Sensor temperatura	1	2€	2€
Sensor Humitat terra	1	2€	2€
Sensor llum	1	2€	2€
Muntatge Complet (hores)	160	40€	6.400€
Alta (hora)	1	45€	45€
Muntatge Parcial (hores)	50	40	2.000€
Suport (tickets)	20	3	60€
Manteniment	12	20	240€
Total			8.987€

Modalitats

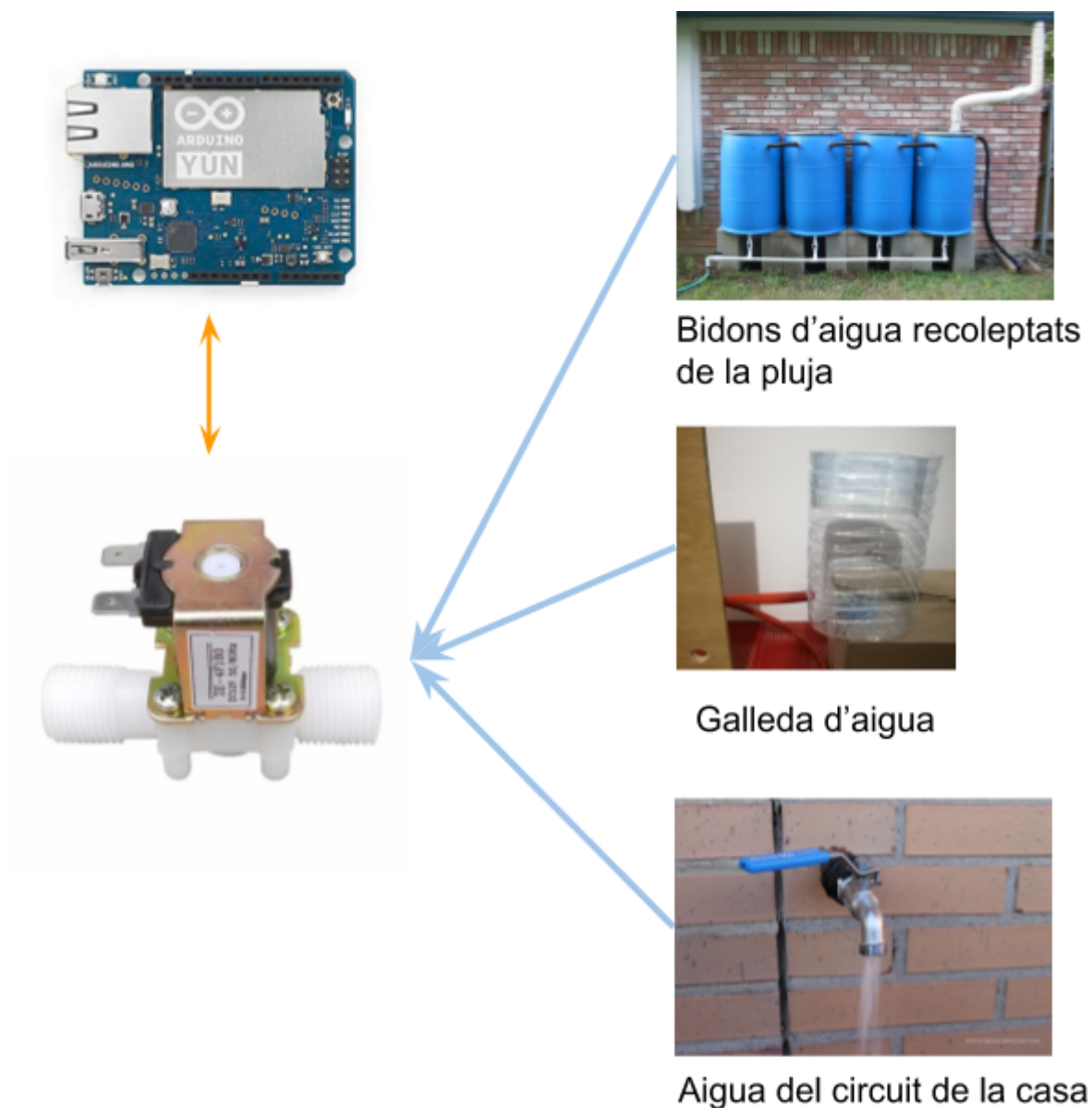
Presupost segons las diferents modalitats

Article	Básic	Muntatge Parcial	Muntatge Complet
Electrovàlvula	x	x	x
Placa Arduino	x	x	x
Sistema de reg	x	x	x
Tubs subministrament aigua	x	x	x
Ordinador	x	x	x
Hosting virtual i Domini	x	x	x
Sensor temperatura	x	x	x
Sensor Humitat terra	x	x	x
Sensor llum	x	x	x
Muntatge Complet (hores)			x
Alta (hora)			
Muntatge Parcial (hores)		x	
Support (tickets)	opcional	opcional	opcional
Manteniment	opcional	opcional	opcional
Total	587€	2.542€	6.942€

Sistema de suministrament d'aigua.

El sministrament d'aigua pot ser multiple. Ja que dependrà de les connexions intermitjes que s'utilitzin, però que també es poden adaptar al tipus de suministrament.

En aquest cas, jo he utilitzat una galleda de 5 litres per fer el projecte. Amb un tub flexible conecat a l'electrovalvula.



Tipus de Reg

El tipus de reg dependrà de la pressió que tingui el sistema de subministrament d'aigua, ja que si utilitzem un sistema de galleda, la pressió que tindrem serà la de la gravetat. I això no ens permetre posar un reg amb aspersion.

Hem de diferenciar en dos sistemes principals de reg, el de superfície i el reg localitzat.

El reg de superfície o per inundació l'aigua es distribueix per tota la superfície del terreny per la gravetat. De forma que el cabal d'aigua del reg disminueix al llarg de l'extensió per infiltració del terreny. Actua sobre les arrels de les plantes, és un sistema poc utilitzat en jardineria per la gran quantitat d'aigua que es necessita i per l'elevada pèrdua d'aigua per evaporació, molt utilitzat a l'hort.

El reg localitzat consisteix a subministrar l'aigua de tal forma que només xopi una part del sol on es troba la planta. Actualment el més utilitzat.

Existeixen diferents tipus de reg localitzats.

Reg per aspersion. Es basa en l'aplicació de l'aigua en forma de pluja. L'aigua és conduïda a pressió fins a arribar a l'aspersion. S'utilitza a superfícies grans, ja que la pressió i el caudal adequats permeten arribar entre 6 i 14 metres de distància. Poden realitzar moviments rotatoris i humidificar zones més o menys circulars de forma uniforme per tota la superfície.



Reg amb difusors. Són semblants als aspersion amb mides més reduïdes. No fan moviments rotatoris, són fixes i s'utilitza també per a superfícies més petites.



Reg gota a gota. Consisteix a aportar aigua de manera localitzada just al peu de la planta. Només es rega la planta o plantes concretes, racionalitzant el reg. Es manté un nivell constant d'humitat del terra i és el que més estalvia.



Cintes d'exsudació. Són canals de material porus que distribueixen l'aigua de manera contínua mitjançant els porus, de tal manera que es forma una línia contínua d'humitat que les fa molt indicades per al reg de plantes en linneà. Hume deixen una gran superfície i és especialment interessant en sols arenosos.



Reg subterrani. Es tracta de tubs perforats que s'enterren en el sol a una determinada profunditat entre 5 i 50 cm segons el que sigui la planta. El funcionament és el mateix que el

reg per gota, la diferencia és que en aquest cas el tub no es veu perquè està enterrada i l'aigua està en contacte directe amb les arrels.



Reg amb mànega. És el reg menys recomanable, ja que es desaprofita molta aigua i no provi una bona uniformitat, a uns llocs cau més aigua, que d'altres. S'utilitza en petites superfícies. Per la gespa, pot ser inadequat però per arbres pot ser el més adequat.



Per tots aquest tipus de reg localitzat, a excepció de reg amb mànega, es pot automatitzar les tasques mitjançant la instal·lació de programadors i electrovàlvules, el que es coneix amb el nom de sistema de reg automatitzat.

Gestió de Riscos

Títol del Risc: Obstrucció del tub o l'electrovalvula amb suministrament de pluja

Probabilitat: 2 - Mitja

Es possible que amb l'aigua de la pluja, a més d'aigua també arrosegui fulles, terra, etc. Això podria fer que s'obstruïssin el tub d'aigua.

Impacte: 3 - Alt

Severitat: $2 \times 3 = 6$ (Alta)

Acció de mitigació: Filtre a la tapa del barril d'aigua.

Les electrovalvules ja porten un filtre, però no estaria de més posar-hi un als bidons que recolecten l'aigua de la pluja.

Probabilitat				
		Baixa	Mig	Alta
Impacte	Baix	1	2	3
	Mig	2	4	6
	Alt	3	5	9

Títol del risc: Sistema electronic exposat a la meteorologia

Probabilitat: 3 - Alta

el sistema està exposat a la meteorologia, vent, pluja, terra, animals,...

Impacte: 3 - Alt

El sistema electronic, està exposat a les condicions meteorologiques i això pot fer que el sistema deixi de funcionar.

Severitat: $3 \times 3 = 9$ (Alta)

Acció de mitigació: Fixa el sistema a terra i protegir-ho amb material impermeable.

S'hauria de fixar a terra i utilitzar una caixa impermeable i dura per protegir-ho dels cops que pogui rebre.

Probabilitat				
		Baixa	Mig	Alta
Impacte	Baix	1	2	3
	Mig	2	4	6
	Alt	3	5	9

Títol del risc: Caiguda de la xarxa d'internet.

Probabilitat: 2- mitja.

Impacte: 2 - mitja

L'impacte sobre el sistema de reg serà mitjà, ja que es deixarà de veure les dades a la plataforma web, i no es podrà interactuar amb l'aixeta. Però el sistema segons els sensors continuarà funcionant, ja que aquest funciona de forma local.

Severitat: $2 \times 2 = 4$ (Mig)

Acció de mitigació: Això dependrà del tipus de caiguda. Si es per la operadora, esperar que es restableixi la connexió. Si es per l'aparell, posar un de més fiable o potent.

Probabilitat				
Impacte		Baixa	Mig	Alta
	Baix	1	2	3
	Mig	2	4	6
	Alt	3	5	9

Títol del risc: Sensor espatllat

Probabilitat: 2- mitja.

Impacte: 3 - Alt

Si un dels sensors, està espatllat això farà que deixi de funcionar el sistema autònom.

Severitat: $2 \times 3 = 6$ (Mig)

Acció de mitigació: Si el valor del sensor no es un número, vol dir que esta donan algún tipus de error. Fer que aquest valor es visualitzi a la web, per a que l'usuari pogui veure que està fallant i el pogui canviar el més aviat possible.

Probabilitat				
Impacte		Baixa	Mig	Alta
	Baix	1	2	3
	Mig	2	4	6
	Alt	3	5	9

Funcionaments tècnics.

El sistema de reg contempla 3 modes. Mode manual, Mode automàtic i repòs.

Per tant, s'ha programat com una màquina d'estats on hi han 3 estats, repòs, manual i automàtic.

D'una banda, el sistema repòs envia cada 60 segons els valors dels sensors a la bbdd de la web thinger.io i regduino.javims.es

L'estat manual envia el valor obert o tancat de l'electrovàlvula quan s'activa o desactiva manualment. Aquest valor d'emmagatzematge a la bbdd per tractar amb la web.

Sensor llum, si el valor és inferior o igual a 400 llavors diu que si es pot amb un 1.
Sensor temperatura, si el valor és superior o igual a 15 graus, diu que si amb un 1.
Sensor humitat, si el valor és inferior a 400, llavors diu que es pot obrir l'electrovàlvula.

Si tots 3 diuen que es pot obrir l'electrovàlvula, llavors s'obre i envia el valor obert a la pàgina web per ser enregistrat. Quan algun dels valors, en diu 0, o que no es pot obrir, llavors l'electrovàlvula es tanca i envia el valor de tancat a la bbdd de la web.

Quan el mode manual entra en acció, el mode automàtic, queda deshabilitat fins que es tanqui manualment l'electrovàlvula.

L'estat automàtic. Llegeix constantment els valors dels sensors i segons si arriben el llindar establert, aquest diuen si es pot obrir o no.

Possibilitats

Automatic				Manual	Repos	Aixeta
Sensor llum	Sensor terra	Sensor Temp				
1	1	1	1	0	0	Oberta
1	0	1	0	1	0	Oberta
0	1	1	0	0	1	Tancat

Cabal d'aigua

La web és capaç de calcular el cabal d'aigua que utilitza per regar la planta. Per saber aquesta informació ens hem basat en els càlculs del temps que ha passat entre l'obertura i el tancament de l'electrovàlvula, el temps que triga n'arriba l'aigua a la planta, i les fórmules de la velocitat i el cabal.

Mides fixes i aproximades

Distància x = 85 cm

Distància y = 40 cm.

Secció tub = 0,31m². El càlcul es fa amb les dades del tub. En aquest cas es un tub de silicona 7mm.

fórmula de la velocitat: $v = x/t$

fórmula del caudal: $Q = v \cdot S$

Q= caudal

v = velocitat

S = secció tub

x = distància.

$$x = \sqrt{85^2 + 40^2} = 93,94 \text{ cm}$$

$$v = 93,94 / 10\text{s} = 9,394 \text{ cm/s}$$

$$Q = 9,394 \text{ cm/s} \cdot 0,31\text{cm}^2 = 2,91214\text{cm}^3/\text{s} = 2,91214\text{e-}6 \text{ m}^3/\text{s}$$

$$Q \text{ total} = Q \cdot \text{temps aixeta oberta}$$

$$Q \text{ total} = 2,91214\text{e-}6 \cdot 60 \text{ segundos} = 0,000175 \text{ m}^3 = 0,000175 \text{ litres en } 60 \text{ segundos.}$$

Per tant si activem manualment l'aixeta, l'haurem de deixar molta estona per deixa anar molta aigua, i si és automàtic, estarà obert fins que el sensor d'humitat arribi el llindar que hem establert com a límit.

Si volem que estigui menys temps obert, doncs haurem de posar un tub amb una secció més gran.

Totes les fórmules que estan fetes amb aproximacions, ja que el terreny és irregular i per a calcular la distància recorreguda s'ha fet el calcula d'un triangle rectangle calculant el valor de la hipotenusa, ja que es coneixien els dos catets.

Plataforma web

Enllaç: <http://regduino.javims.es>

S'obtarà per dues plataformes, ja que una plataforma personalitzada en un temps concret pot tenir problemes de seguretat importants i això pot fer que persones no autoritzades puguin fer manipulacions, en aquest cas de l'aixeta, i fer mal. D'aquesta manera, s'ha fet una web personalitzada per a mostrar la informació dels sensors de la placa arduino i una altre plataforma que es diu thinger.io, per a la manipulació manual de l'aixeta, tot i que també es poden connectar els sensors per visualitzar les dades.

Plataforma Regduino

La plataforma personalitzada consta de diferents capes.

Frontend

Aquesta capa té tot la part visual de l'aplicació, i la podem resumir en 4 fitxers.

index.php > main principal que fa un include del fitxer funcions.php

funcions.php > cada part de la pàgina (menu, capçalera, peu, contingut,...) es troba a una funció dins d'aquest.

javascript.js > conté les accions que fa l'usuari quan fa click a les diferents part de la pàgina. O l'actualització dels valor cada 3 segons per exemple.

style.css > tota la part de maquetació, colors, estils, ...

Backend

Aquesta part es la que no es veu visualment, però forma part de les dades.

ajaxfunciones.php > fitxer que es crida desde arduino i rep els valor dels sensors.

la crida es realiza de la següent manera: `ajaxfunciones.php?sensor=1&valor=13`

readValue.php > aquest fitxer ens serveix per llegir un valor d'un sensor.

la crida es fa de la següent manera: `readValue.php?sensor=terra`

Database.php > es una clase que es conecta a la BBDD.

Datosbbdd.php > es una clase que conté les sentencies sql que s'utilitzaran per agafa dades.

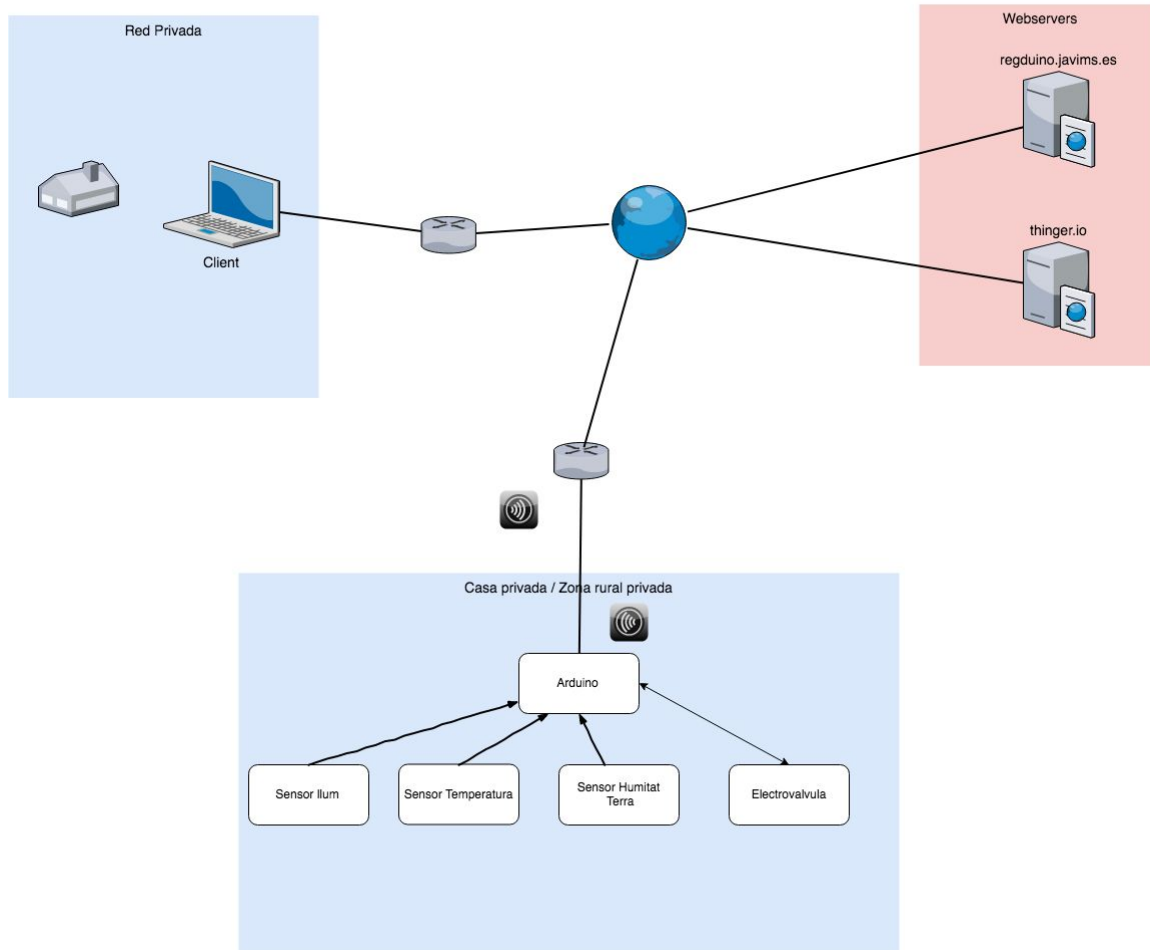
Tenim una carpeta datos on hi han els fitxers que s'utilitzen per mostrar els valors reals al moment. Aquests fitxers s'actualitzen quan es fa la crida del fitxer **ajaxfunciones.php**.

Diagrama de Conexió

Diagrama que mostra la connexió física per a l'ús d'aquest sistema.

El client es l'ordinador de connexió de l'usuari, qui serà el que es connecti als webservers per interactuar amb arduino.

Conexio de dispositius



Casos d'ús

Aquests son els casos d'ús que podem trobar al sistema.

Cas d'ús: Obrir aixeta

Actor: usuari, client, etc

Tipus: Esencial

Precondició: Aixeta tancada

Postcondició: L'aigua sortirà i regarà la planta, després d'un temps es tancarà

Proposit: Que es regi la planta.

Resum: L'usuari apretarà un boton desde la plataforma web i això farà que s'obri l'aixeta per a que l'aigua pogi regar la planta.

Cas d'ús: Tancar aixeta

Actor: usuari, client, etc

Tipus: Esencial

Precondició: Aixeta oberta

Postcondició: L'aigua deixarà de sortir.

Proposit: que la planta no tingui més aigua de la que necessita.

Resum: L'usuari apretarà un boton desde la plataforma web i això farà que tanqui l'aixeta perque creu que ja es suficient l'aigua que té la planta.

Cas d'ús: Registrar temperatura

Actor: Arduino

Tipus: Esencial

Precondició: Arduino conectada a la xarxa

Postcondició: valor sensor registrat

Proposit: Registrar la temperatura a una base de dades per saber quina temperatura feia en aquell moment.

Resum: Arduino enviarà el valor de la temperatura atraves d'internet a la web i l'enregistrerà a la base de dades.

Cas d'ús: Registrar humitat terra

Actor: Arduino

Tipus: Esencial

Precondició: Arduino conectada a la xarxa

Postcondició: valor sensor registrat

Proposit: Registrar la humitat del terra a una base de dades per saber quina temperatura feia en aquell moment.

Resum: Arduino enviarà el valor de la humitat de la terra a traves d'internet a la web i l'enregistrerà a la base de dades.

Cas d'ús: Registrar Llum

Actor: Arduino

Tipus: Esencial

Precondició: Arduino conectada a la xarxa

Postcondició: valor sensor registrat

Proposit: Registrar la llum a una base de dades per saber si es de dia o de nit en aquell moment

Resum: Arduino enviarà el valor de la llum a través d'internet a la web i l'enregistrerà a la base de dades.

Cas d'us: Consultar els valor dels sensors en directe

Actor: usuari, client

Tipus: Esencial

Precondició: connectar-se a la pàgina web on estan els valor

Postcondició: visualitzar els valors.

Proposit: Consultar els valors dels sensors en directe, per conèixer l'estat de la planta.

Resum: L'usuari entrarà a la web del sistema i a la pestanya on posar sensors, podrà veure quins valors tenen.

Cas d'us: Consultar els valor dels sensors registrats.

Actor: usuari, client

Tipus: Esencial

Precondició: connectar-se a la pàgina web on estan els valor

Postcondició: visualitzar el registre.

Proposit: Consultar els valors dels sensors registrats.

Resum: L'usuari entrarà a la web del sistema i a la pestanya on posar registre, podrà veure quins valors estan registrats.

Cas d'us: Escollir el sensor a consultar

Actor: usuari, client

Tipus: Esencial

Precondició: connectar-se a la pàgina web on estan els valors registrats

Postcondició: visualitzar els valor del sensor escollit.

Proposit: Seleccionar un sensor per visualitzar els valors registrats d'aquell sensor.

Resum: L'usuari entrarà a la secció de registre i podrà seleccionar quin sensor vol visualitzar.

Aquest diagrama ens permet saber quins usos es poden donar a l'hora d'utilitzar l'aplicació. Trobem que per part de l'usuari, pot fer obrir tancar l'aixeta, consultar dades enregistrades i dades en directe. I trobem un cas d'ús que no ho fa l'usuari, sino la placa arduino, utilitza el servidor per enregistrar les dades.

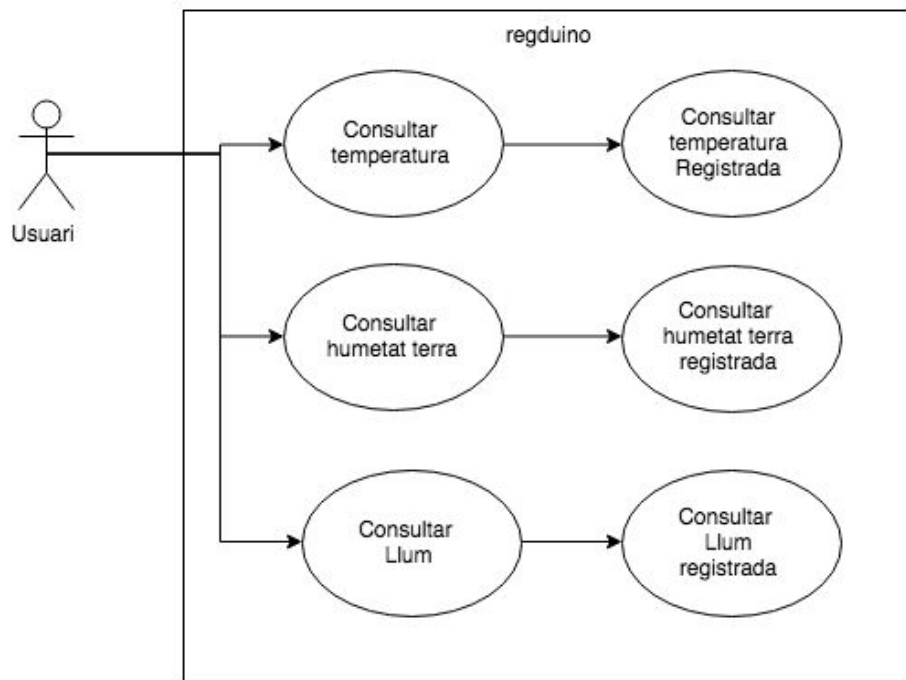
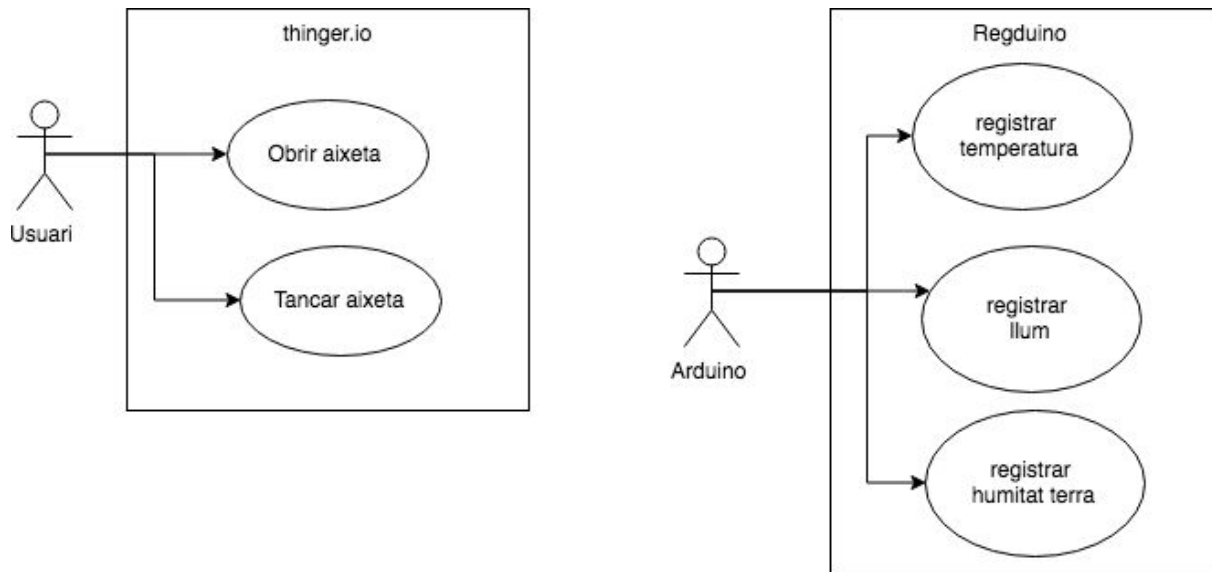


Diagrama d'Estats dels sensors

El següent es el diagrama d'estat dels sensors. Amb aquest diagrama mostrem els estats en que podem trobar els sensor connectats a la placa arduino.

Aclarir que la electrovalvula estarà sempre en estat tancat, només obrirà quan les condicions dels altres sensors connectats a la placa o permetin o per acció de l'usuari.

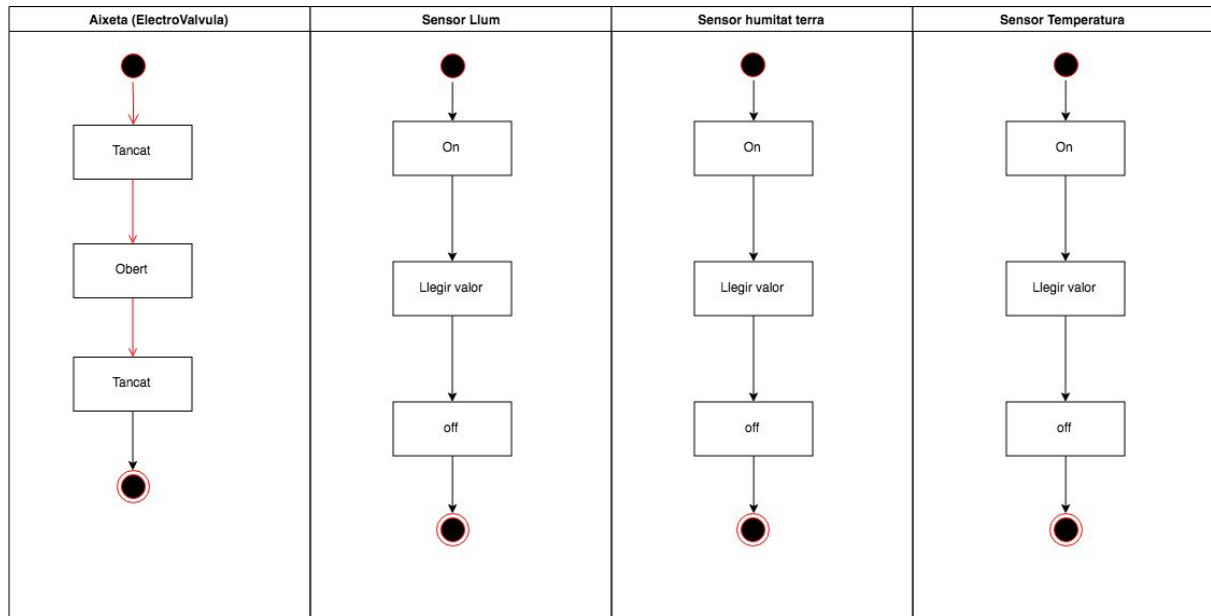


Diagrama de flux dels sensors

Diagrama que ens mostra el flux del codi del firmware d'arduino. Aquest ens permet saber quan s'accionarà l'electrovalvula, segons les condicions de cada sensor i quines seran aquestes condicions.

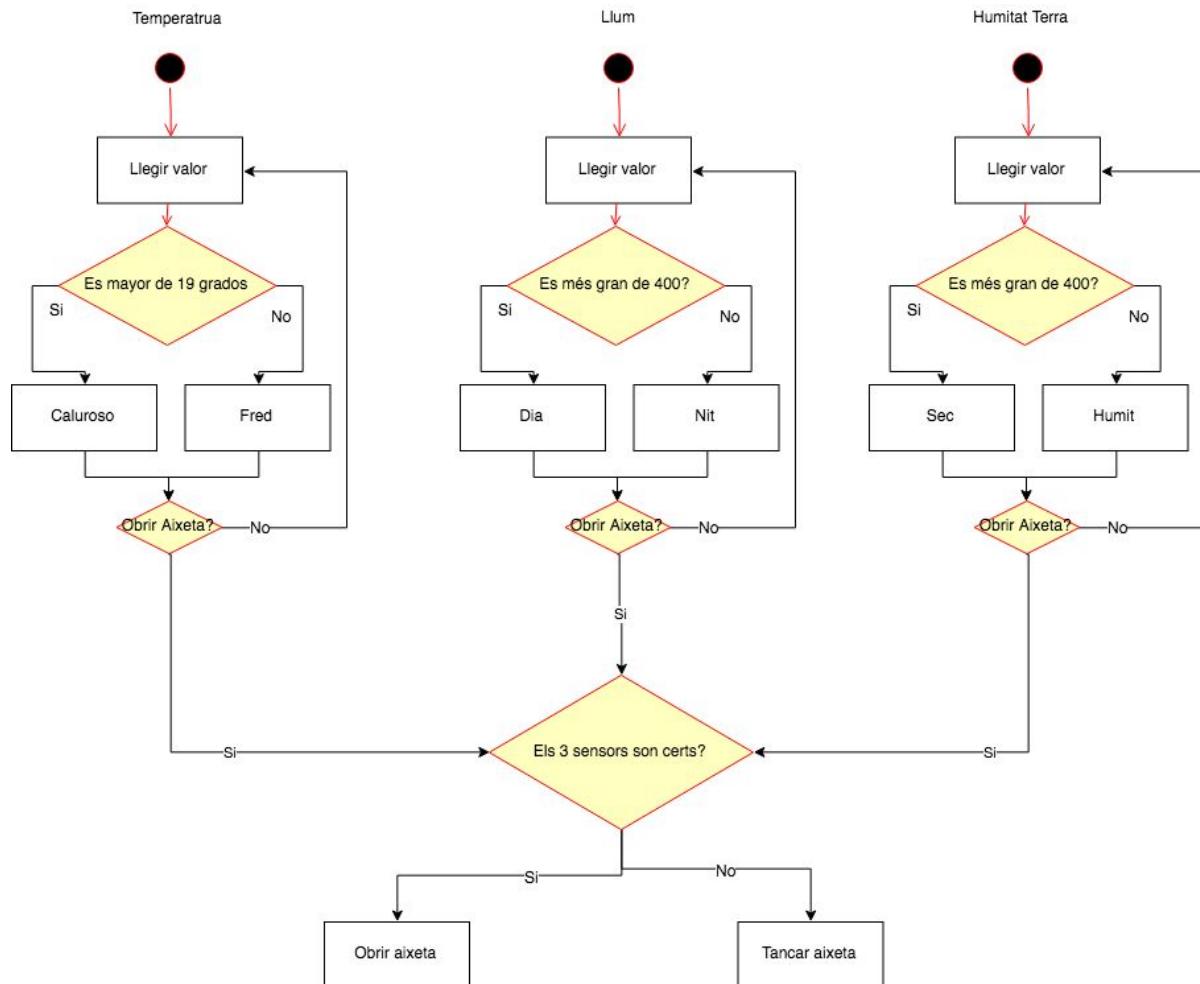
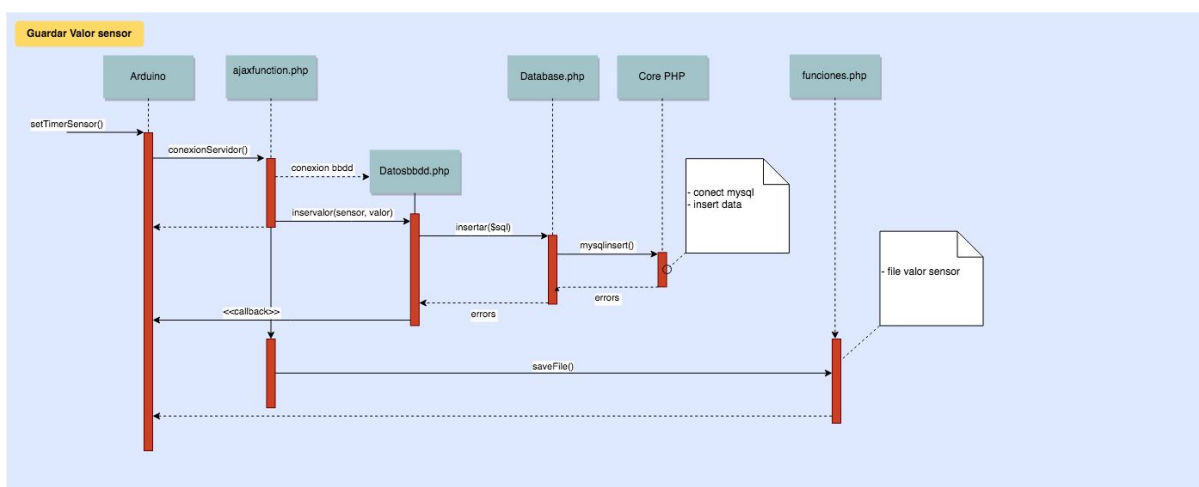


Diagrama de seqüència

Diagrama de seqüència. Ens permet obtenir una visió global del codi entre l'arduino i la plataforma web. Ja que hi ha una seqüència desde que arduino envia el valor al servidor, i que fa el servidor amb aquest valor que rep.

Si ens fixem en el diagrama, quan arduino emet el valor, hi han dos seqüències paral·leles. Guardar a la base de dades i a un fitxer.

Això es perque l'enregistrament a la base de dades es per tenir un historic i en un fitxer només guarda l'últim valor, per a la visualització en directe del valor a la web.



Pantallas

<http://regduino.javims.es>

Aquesta pàgina té 4 pantalles que explicaré a continuació:

Pantalla principal

Aquesta pantalla ens mostra una petita descripció del sistema arduino.



Regduino

Inici Sensors **Registre** About

Sistema de Reg per Arduino

Regduino, és un sistema de reg autogestionat, que consisteix en què una planta auto reguli l'aigua que necessita per alimentar-se.

Aquest auto gestió l'aconsegurem amb l'aigua de la pluja, on s'anirà omplint un recipient, d'aquesta manera tindrem acumulada aigua que després anirem subministrant.

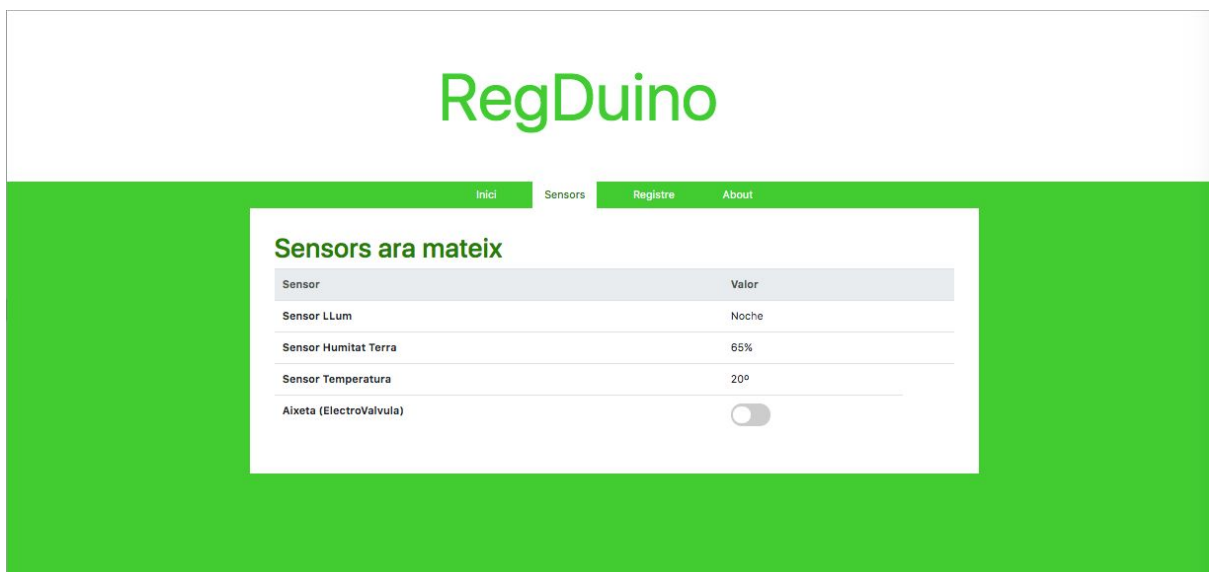
Aquest subministrament es farà en funció de diferents valors, que els aconseguirem segons uns sensors que instal·larem a la plantació, per agafar la humitat de la terra, la temperatura, la llum, ... Aquests ens donaran la informació que necessitem per poder anar subministrant l'aigua en funció de si necessita o no necessita la plantació.

D'altra banda, com que és necessari alimentar la placa arduino, que és la que processara totes les dades que ens arribin, s'utilitzarà una placa solar per alimentar aquesta.



Pantalla sensors

Aquesta pantalla ens mostra els valor dels sensors en directe. Es refresca cada 3 segons, i mostra l'últim valor emès per la placa arduino.



Regduino

Inici Sensors **Registre** About

Sensors ara mateix

Sensor	Valor
Sensor LLum	Noche
Sensor Humitat Terra	65%
Sensor Temperatura	20°
Aixeta (ElectroValvula)	<input type="checkbox"/>

Pantalla registre

A aquesta pantalla podem veure el registre dels sensors. Podem trobar un combobox, on podem escollir de quin sensor volem veure les dades enregistrades.

Per seguretat en l'enregistrament i no generar molts valors, s'ha posat una condició per agafar els valors en 3 moments del dia, d'aquesta manera no enmagatzamanet molta informació. Però això segons les necessitats pot variar.



Sensor	Valor	Data/Hora
Temperatura	20	2019-05-22 17:23:02
Temperatura	20	2019-05-22 16:55:20
Temperatura	19	2019-05-21 18:46:43
Temperatura	19	2019-05-21 18:46:33
Temperatura	19	2019-05-21 18:46:24
Temperatura	19	2019-05-21 18:46:15
Temperatura	19	2019-05-21 18:46:06
Temperatura	19	2019-05-21 18:45:57

Pantalla About

Aquesta pantalla es sobre qui l'ha fet, amb la col·laboració.



By Javier Molina Suárez
Treball final de Grau per a la Universitat Oberta de Catalunya (UOC)

Plataforma Thinger.io

Introducció

Per aquest projecte, vist les dificultes que vaig trobar per poder fer la interacció d'obrir i tancar l'electrovalvula amb arduino per la propia plataforma, he optat per fer ús d'aquesta plataforma especialitzada en amb l'IOT.

Per decidir a fer ús d'aquesta plataforma, vaig fer una investigació per la xarxa, quines plataformes existien per la IOT. Amb aquesta cerca, vaig trobar sentilo, thinger.io i foros que explicaven com obrir i tancar amb la teva propia plataforma.

Vaig descartar la opció de fer el meu propi sistema, ja que vaig trobar un inconvenient important, que es la seguretat, que no contemplava amb aquest projecte.

Es important tenir en compte, que qualsevol interacció ha de comptar amb una seguretat a l'hora de treballar desde internet i això estava fora d'aquest projecte.

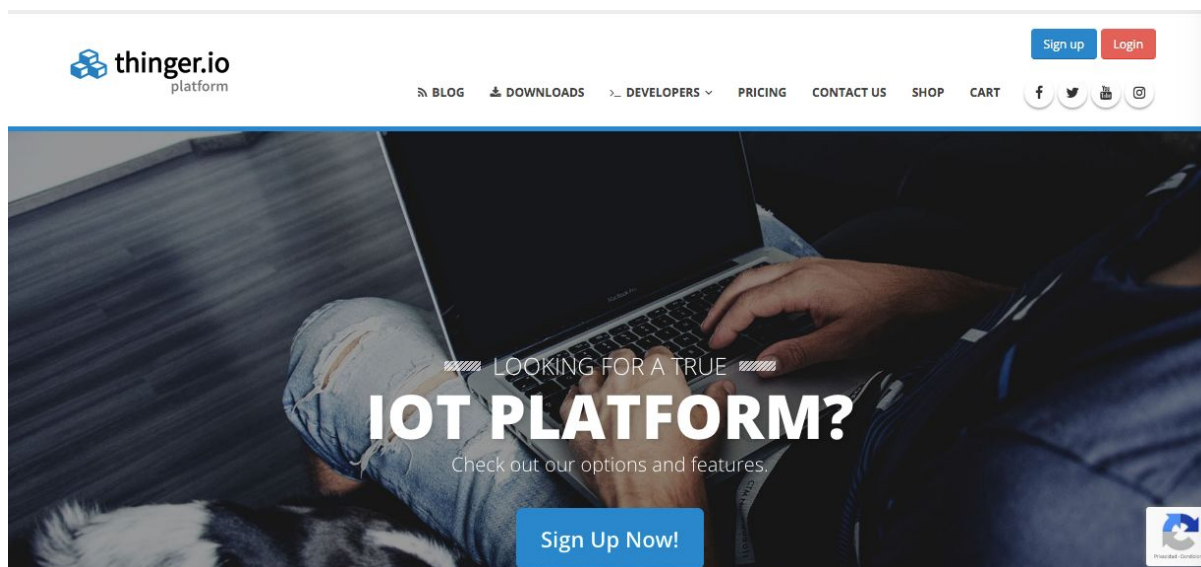
Una altre plataforma interessant va ser SENTILO, desenvolupada desde la diputació de barcelona i la maquina virtual acceisble per fer el teu propi servidor. Però vaig trobar un inconvenient, que es una plataforma especialitzada en la recepció de les dades, però encara queda una mica per a la interacció actuadors amb arduino yun.

D'aquesta manera, hem vaig decantar per thinger.io, ja que et pots donar d'alta a la web que tenen i configurar-ho amb la placa que tinguis, posant les credencials corresponents.

Funcionament

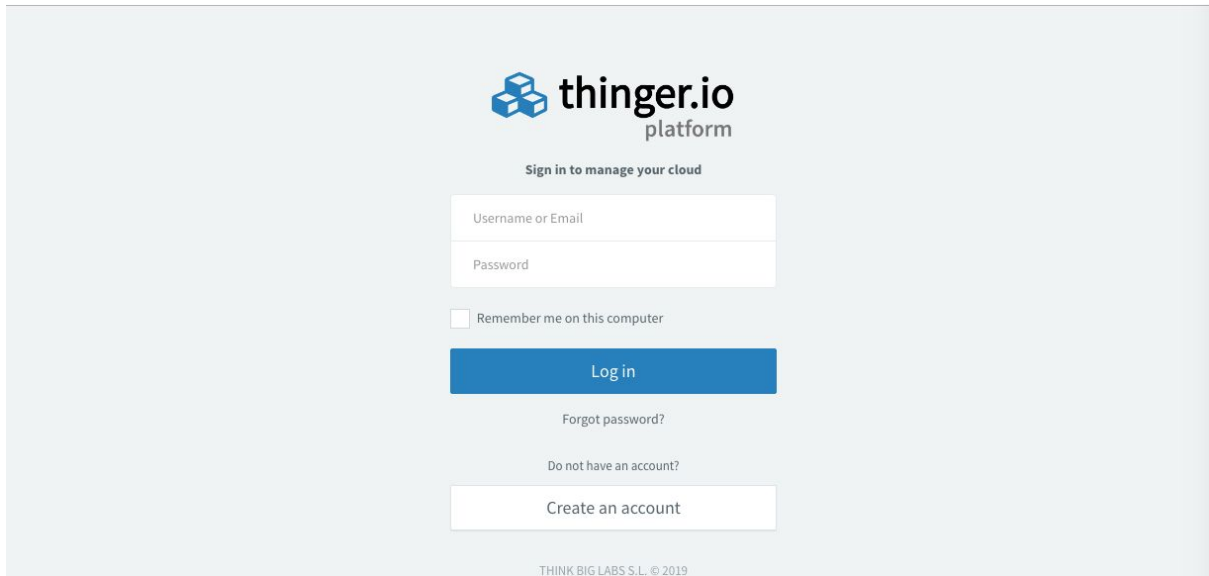
Pantallas

Pantalla principal



Pantalla login thinger.io

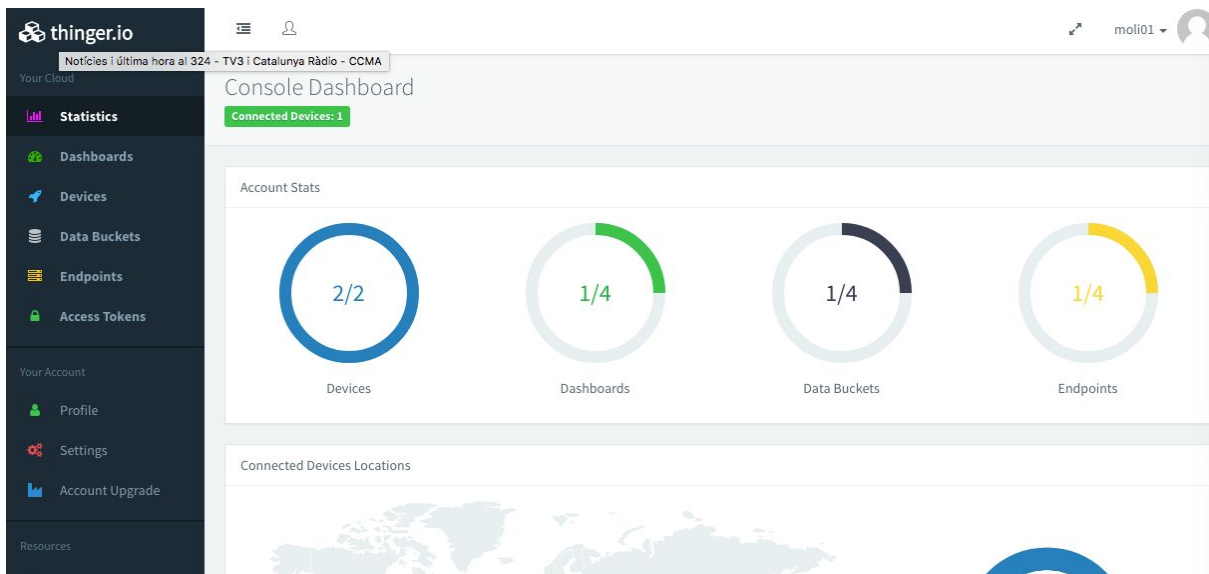
Amb aquesta pantalla ens podem registrar o accedir si ja ho hem.



The screenshot shows the login page for thinger.io. At the top center is the logo, which consists of three blue cubes forming a larger cube, followed by the text "thinger.io" in a bold, sans-serif font, and "platform" in a smaller font below it. Below the logo is the text "Sign in to manage your cloud". There are two input fields: "Username or Email" and "Password". Below these is a checkbox labeled "Remember me on this computer". A prominent blue button labeled "Log in" is centered below the checkbox. Underneath the button are two links: "Forgot password?" and "Do not have an account?". At the bottom of the form is a white button labeled "Create an account". At the very bottom of the page, in small text, it says "THINK BIG LABS S.L. © 2019".

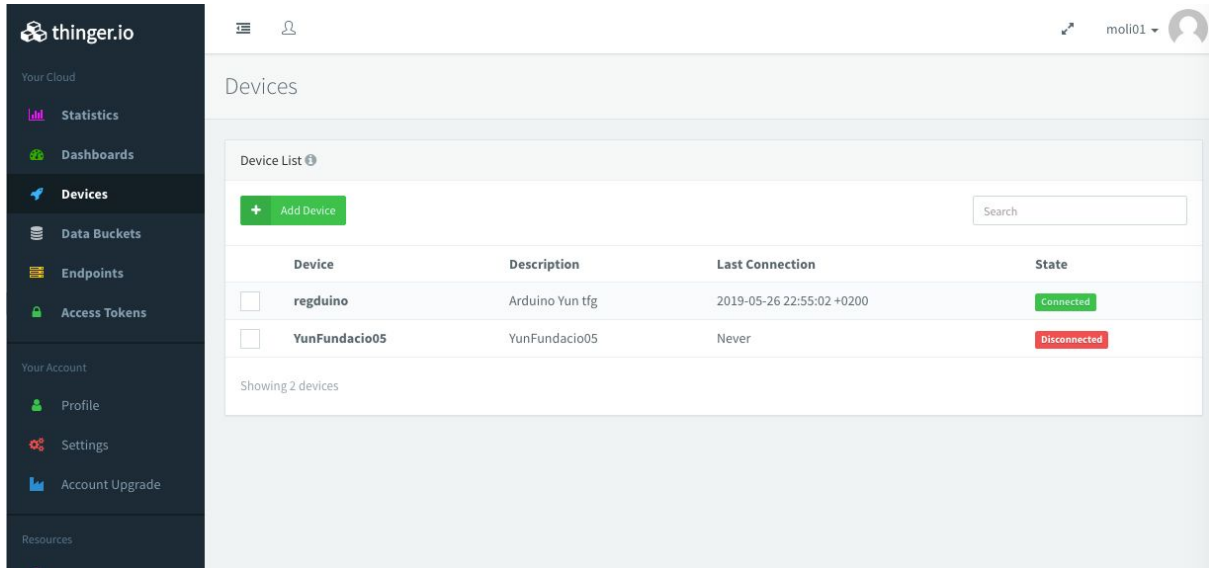
Pantalla Dashboard (un cop logat)

Només entrar trobem les estadístiques del que tenim connectat i creat al nostre compte.



Pantalla del llistat dels dispositius connectats.

Aquesta pantalla tenim tots els dispositius que podem connectar a la placa. I ens diu si estan connectats o no.

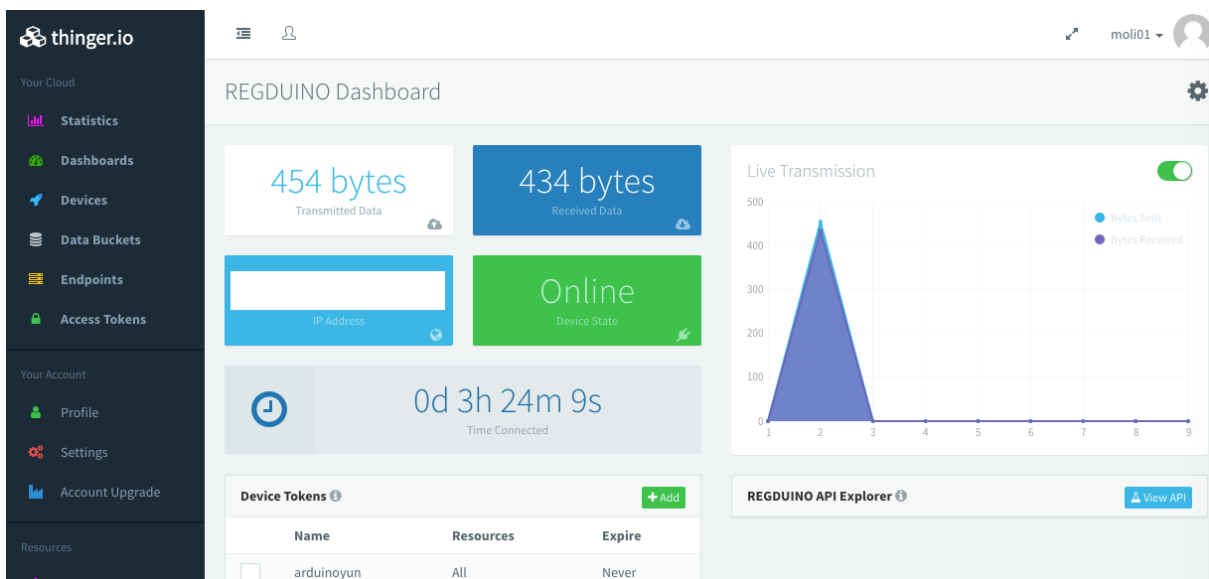


The screenshot shows the 'Devices' page in the Thinger.io interface. The left sidebar contains navigation options like 'Statistics', 'Dashboards', 'Devices', 'Data Buckets', 'Endpoints', 'Access Tokens', 'Your Account', and 'Resources'. The main content area is titled 'Devices' and features a 'Device List' section with an 'Add Device' button and a search bar. Below this is a table with the following data:

Device	Description	Last Connection	State
<input type="checkbox"/> regduino	Arduino Yun tfg	2019-05-26 22:55:02 +0200	Connected
<input type="checkbox"/> YunFundacio05	YunFundacio05	Never	Disconnected

Below the table, it says 'Showing 2 devices'.

Pantalla estadístiques del Dispositiu connectat



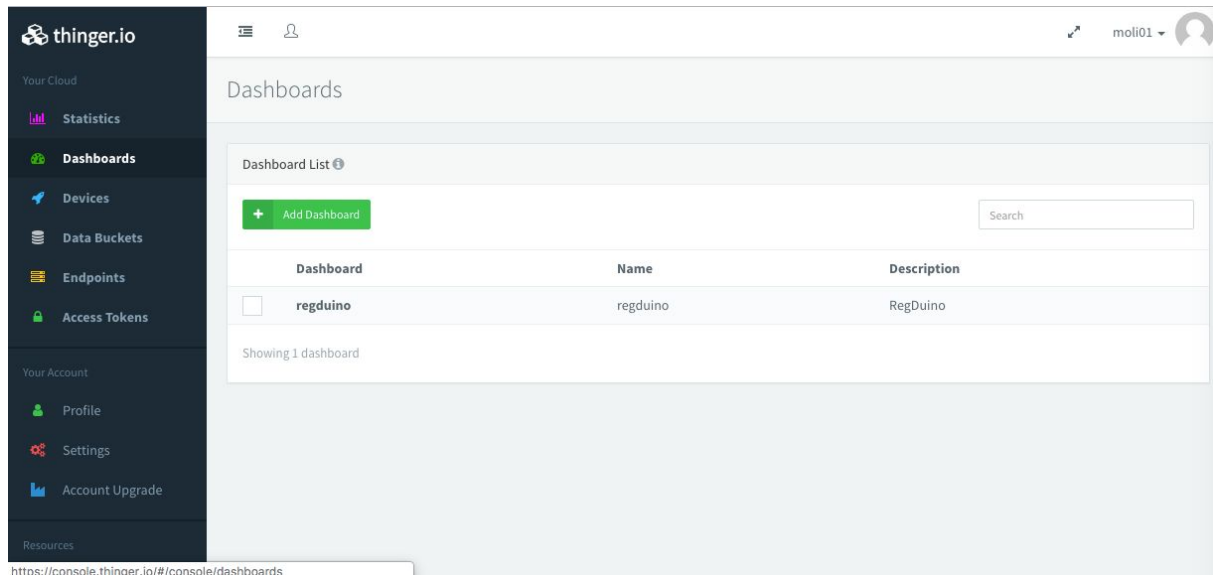
The screenshot shows the 'REGDUINO Dashboard' in the Thinger.io interface. The left sidebar is the same as in the previous screenshot. The main content area is titled 'REGDUINO Dashboard' and features several widgets:

- Transmitted Data:** 454 bytes
- Received Data:** 434 bytes
- IP Address:** [Input field]
- Device State:** Online
- Time Connected:** 0d 3h 24m 9s
- Live Transmission:** A graph showing data transmission over time. The y-axis represents bytes (0 to 500), and the x-axis represents time (1 to 9). The graph shows a peak of approximately 450 bytes at time 2. A legend indicates 'Bytes Sent' (blue) and 'Bytes Received' (purple).
- Device Tokens:** A table with the following data:

Name	Resources	Expire
<input type="checkbox"/> arduinoyun	All	Never

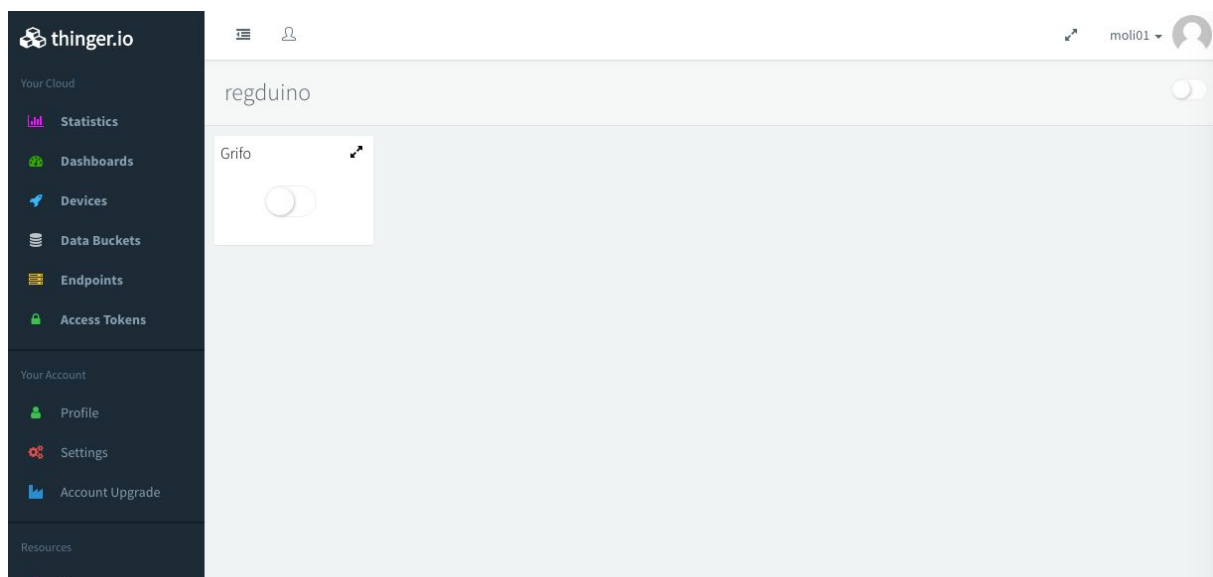
At the bottom right, there is a 'REGDUINO API Explorer' widget with a 'View API' button.

Dashboard



Widgets per la interacció amb la placa (dins del dashboard)

Dins d'aquesta pantalla tenim els diferents widgets per interactuar amb la placa, o rebre la informació dels sensors connectats.



Firmware per Arduino Yun

```
#define _DISABLE_TLS_
#define _DEBUG_

#include <Thingeryun.h>
#include <BridgeSSLClient.h>
#include <dht11.h>

#define USERNAME "user_name"
#define DEVICE_ID "device_id"
#define DEVICE_CREDENTIAL "device_credenciales"
#define PINGRIFO 11
#define PINTEMPORATURA = 8;

Thingeryun thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);
dht11 DHT;

void setup() {
  pinMode(PINGRIFO, OUTPUT);

  // initialize bridge
  Bridge.begin();

  // pin control example (i.e. turning on/off a light, a relay, etc)
  thing["grifo"] << digitalPin(PINGRIFO);
  // resource output example (i.e. reading a sensor value, a variable, etc)
  thing["millis"] >> outputValue(millis());
}

void loop() {
  thing.handle();
}
```

Sistema arduino

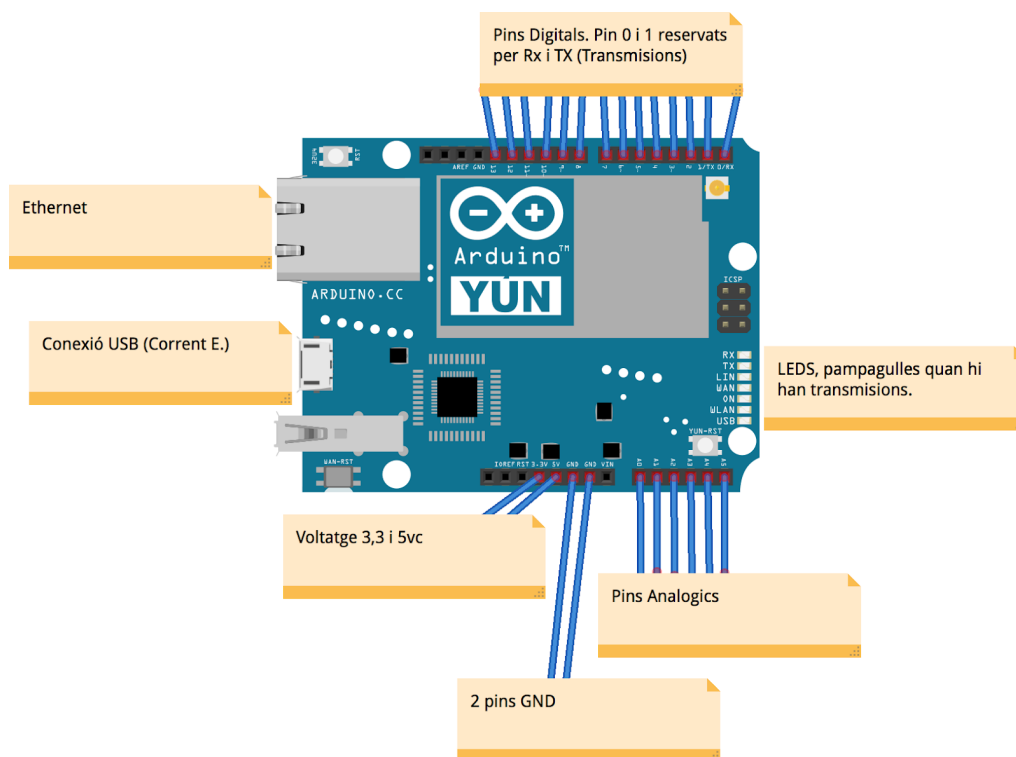
Arduino YUN.



Descripció

He escogit aquesta placa, ja que tota té tota la integració de les xarxes (ethernet i wifi) feta i només s'ha de programar l'enviament de dades i no la configuració de les connexions. També incorpora una ranura de tarjeta micro sd per a la persistència de dades. A més incorpora un servidor propi desde on es pot configurar moltes més coses.

Disseny connexions



fritzing

Codifont

Per programar la placa Arduino amb els sensors. Hem de partir de la base d'aquest dos conceptes:

```
void setup(){}
```

Aquesta ens permet configurar amb valors inicials la placa arduino i que s'executaràn nòmés un cop, quan encenem la placa arduino.

```
void loop(){}
```

Aquesta es un bucle infinit, es repeteix l'execució quan arriba al final del codi. Aquesta funció està sincronitzada amb el temps, per tant podrem dir que un moment determinat ens envii els valor dels sensors, o ens el mostri per pantalla.

També hem d'incloure llibreries al nostre codi ja que necessitem utilitzar aquestes llibreries, per poder accedir a les funcions que llegeixen el valor del sensor i el serveixen.

O funcions per poder connectarnos a la xarxa per enviar les dades.

Al nostre codi té les següents inclusions de les llibreries que necessitem:

```
#include <dht11.h> //Aquesta la necessitem per recórig el valor de la temperatura
#include <Servo.h>
#include <Bridge.h> //Conexió pont entre el codi i la conexio a la xarxa de Arduino Yun
#include <Console.h> //Mostra els valors per consola quan ens conectem via wifi
#include <HttpClient.h> //Per a la conexió http
#include <BridgeClient.h> //Conexió pont entre el codi i la conexio a la xarxa de Arduino Yun
#include <BridgeServer.h> //Conexió pont entre el codi i la conexio a la xarxa de Arduino Yun
#include <Process.h> //Tracament amb les dates
#include <SimpleTimer.h> //Ens servirà per a la declaració inicial de timers.
```

També hem de declarà altres elements com les variables amb el valor del pin que utilitzarem:

Declaració inicial d'objectes com timers i dht11

```
dht11 DHT; //Note: DHT on behalf of the DHT11 sensor
Process date; //Para la fecha
SimpleTimer timerTemperatura;
SimpleTimer timerSensorTierra;
SimpleTimer timerEV;
SimpleTimer timerLuz;
```

Declaració PINs

```
const int dht11_data = 8; //Please put the DH11's dht11_data pin connect with arduino digital Port 6
const int sensorTierra = A0;
const int sensorTouch = 12;
const int pinSensorLuz = A1; //Sensor de luz
const int pinElectroValvula = 11; // pinElectroValvula
```

Declaració Variables

```
boolean shtcanopenev, slcanopenev, stcanopenev, openmanual;

int temp=0;
int openev=0;
int hum=0;
int activatedMotor=0;
```

```
int valueLuz = 0;  
String cc;  
String valorConsole;
```

Envio datos a servidor

```
IPAddress server(91,146,101,244);  
BridgeClient client;  
String parametri = "";
```

Sensor humitat

Descripció

Aquest sensor es el que va la maceta. Aquest agafarà la lectura de si el terra es troba sec o humitat i mitjançant aquest el programa tindrà un factor per valorar si s'ha d'obrir o no l'aixeta.

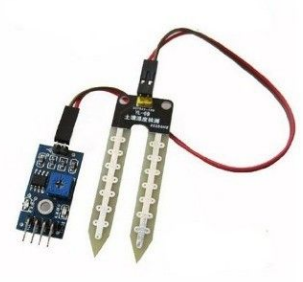
Per connectar-ho a la placa utilitzem 3 cables, GND(terra), VCC(voltatge) i D0 (digital). Te un quart pin que es per connectar-ho al pin analogic de la placa.

Els valors que recullirà seràn:

0 -> h m d.

1023 -> sec.

La condici  al codi es que si el valor  s m s gran que 400, la terra necessitar  aigua.

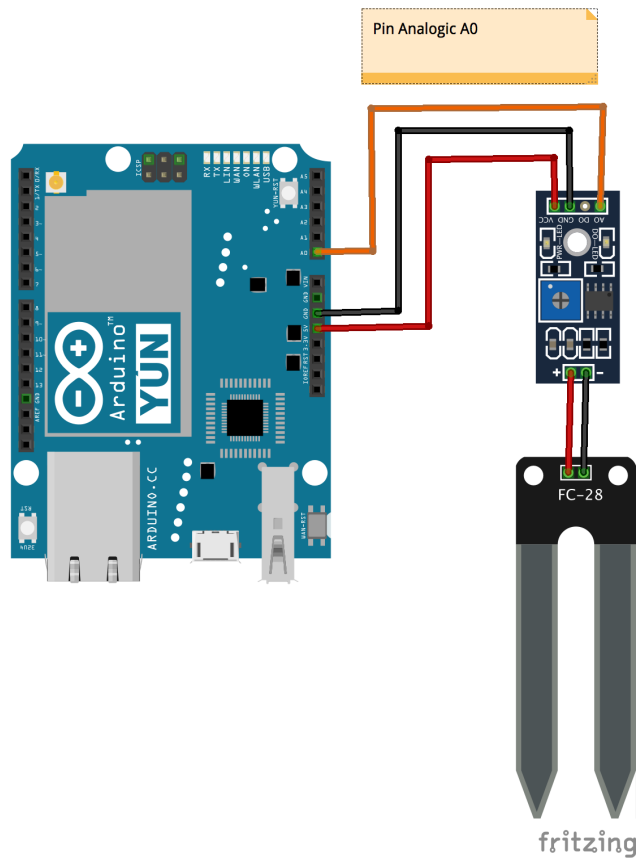


Disseny conexi 

Taronja: Dades

Negre: GND

Vermell: 5v



Codifont

```
void sensorHumedadTierra(){
  int humedad = analogRead(sensorTierra);
  Serial.print("Humedad Tierra=");
  Serial.println(humedad);
  Console.print("Humedad Tierra=");
  Console.println(humedad);

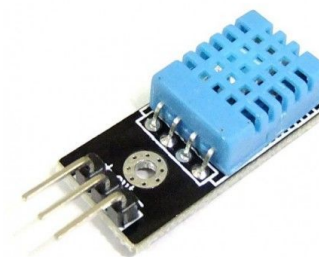
  if(humedad>=400){
    Console.println("Sensor humedad tierra la electrovalvula se puede abrir");
    shtcanopenev = 1;
  }else{
    shtcanopenev = 0;
    Console.println("Sensor humedad tierra la electrovalvula NO se puede abrir");
  }

  Console.print("SensorTierra:");
  Console.println(shtcanopenev);
}
```


Sensor temperatura

Descripció

Sensor utilitzat és el DHT11, que aquest sensor agafa la temperatura i la humitat ambiental de la zona. Per tant tenim dos factors més per a tenir en compte a l'hora d'obrir l'aixeta.



Els valors que recullirem d'aquest sensor, seràn:

Temperatura i humitat.

Per determinar si la valvulà s'obrirà, només necessitarem la temperatura.

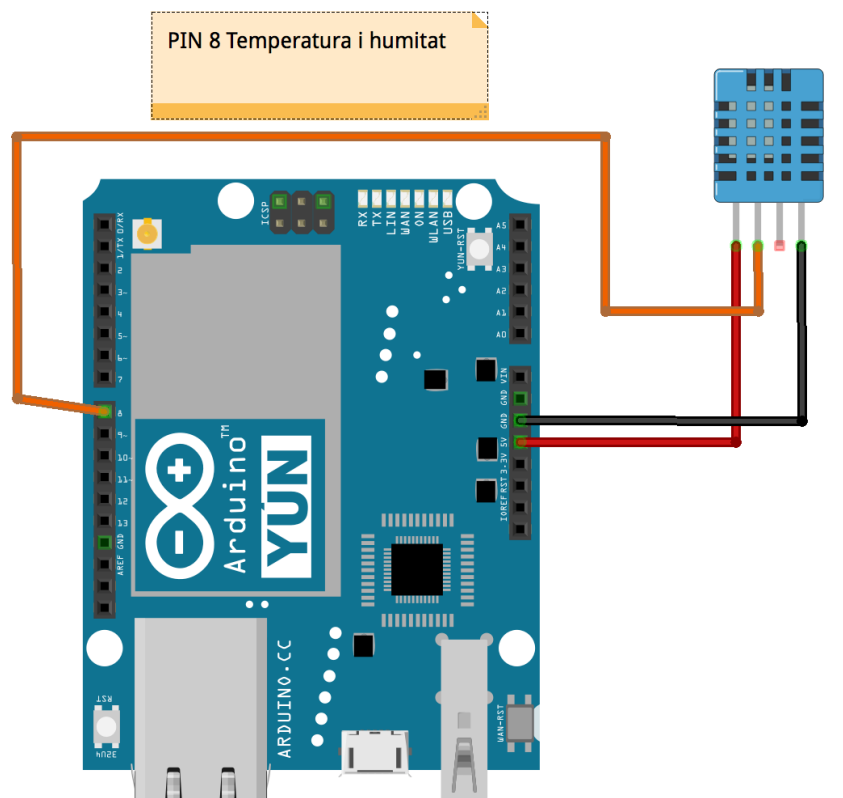
He posat una temperatura de 19°C. Ja que a l'hivern com que la humitat es més alta trigarà més a assecar-se el terra.

Disseny connexió

Taronja: Dades

Negre: GND

Vermell: 5vc



fritzing

CodiFont

```
void SensorTemperatura(){

    DHT.read(dht11_data);
    temp=DHT.temperature;
    hum=DHT.humidity;

    //Console.println(temp);
    Console.print("Hum=\t%");
    Console.print(hum);
    Console.print("\tTemp=\t%");
    Console.println(temp);
    //delay(3000);
    Serial.print("Hum=\t%");
    Serial.print(hum);
    Serial.print("\tTemp=\t");
    Serial.println(temp);

    if(temp >=19){
        stcanopenev = 1;
        Console.println("Sensor Temperatura la electrovalvula se puede abrir");
    }else{
        stcanopenev = 0;
        Console.println("Sensor Temperatura la electrovalvula NO se puede abrir");
    }

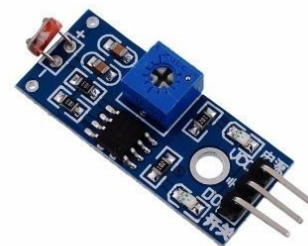
    Console.print("SensorTemperatura:");
    Console.println(stcanopenev);

}
```

Sensor Llum

Descripció

Aquest sensor es una fotoresistencia. Es el que determinarà l'ausència de llum per al suministrament de l'aigua, ja que per poder regar les plantes es millor fer-ho quan no hi ha llum ja que la llum solar pot fer que a través de les gotes d'aigua faci l'efecte lupa.



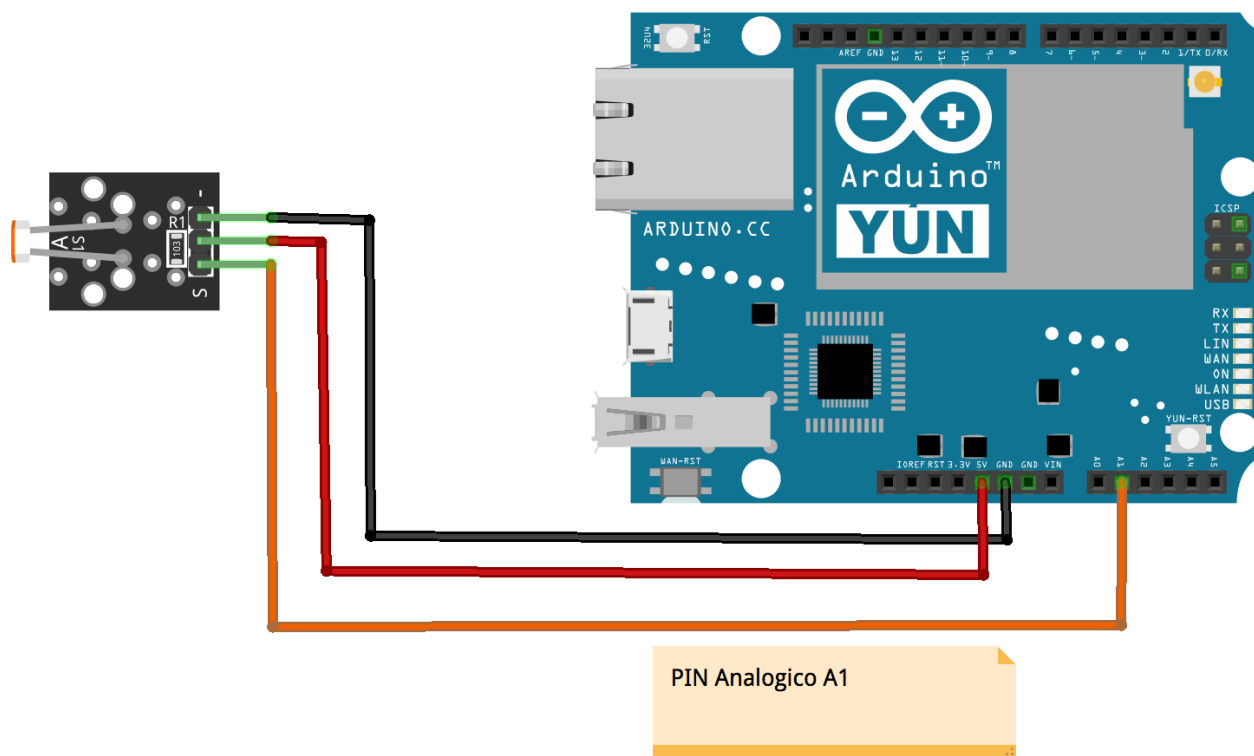
El valor que recollim es un valor analògic, per tant, quan més alt es el valor, més llum rep el sensor. Com que l'important es que no es regui quan hi ha sol directe. He posat un valor inferior a 400, ja que es pot regar amb una mica de llum, mentre no sigui un valor alt.

Disseny conexió

Taronja: Dades

Negre: GND

Vermell: 5vc



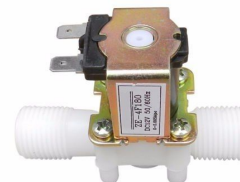
Codifont

```
void sensorLuz(){  
  
    valueLuz = analogRead(pinSensorLuz); // Pull High Restance  
    Console.print("Valor Luz: ");  
    Console.println(valueLuz);  
  
    if (valueLuz < 400){  
        //Terminar de ajustar  
        Console.println("Es de Dia");  
        slcanopenev = 0;  
    }else{  
        Console.println("Es de Noche");  
        slcanopenev = 1;  
        Console.println("Sensor luz, la electrovalvula se puede abrir");  
    }  
  
    Console.print("SensorLuz:");  
    Console.println(slcanopenev);  
  
}
```

Electro Valvula

Descripció

Mecanisme elèctric. Es el que ens permet o no el pas de l'aigua. Necessitarà corrent d'un endoll per poder funcionar, però amb un transformador de 12v. ja que es l'energia que necessita per a funcionar.



A més per poder fer-lo servir amb arduino necessitarem també un Relay de 5vc on connectarem la placa arduino i la electro valvula.

Dades tècniques

Material: Latón

Voltaje: 12V

Diámetro nominal: 1/2 "

Presión de agua: 0.02-0.8Mpa

Tamaño: 6.5 x 6 x 3.5 cm / 2.56 x 2.36 x 1.38 pulgadas

Peso: 218g

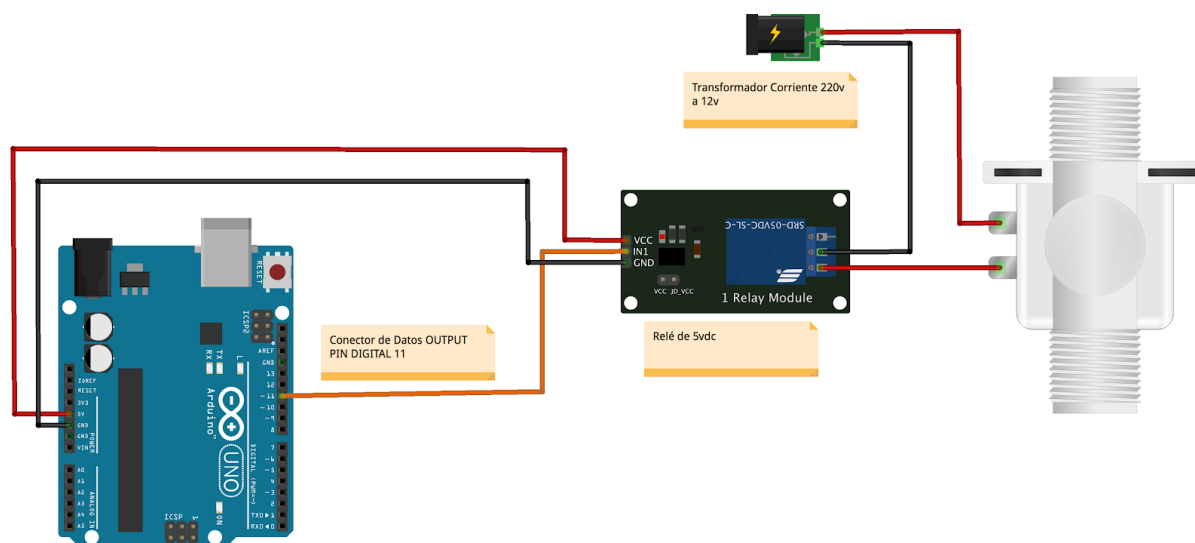
Tipo: guía (unidireccional) válvula solenoide

Disseny conexió

Taronja: Dades

Negre: GND

Vermell: 5vc



Codi font

```
void electroValvula(){
  //digitalWrite(pinElectroValvula,LOW);
  //Console.print("Estado Valvula:LOW");
  //Console.print("Estado Valvula:");
  if(openev==0){
    digitalWrite(pinElectroValvula,HIGH);
    openev=1;
    delay(5000);
    //Console.println("Abierta");
    digitalWrite(pinElectroValvula,LOW);
    regado=1;
  }else{
    digitalWrite(pinElectroValvula,LOW);
    openev=0;
    //Console.println("Cerrada");
  }

  if(shtcanopenev == 1 and slcanopenev == 1 and stcanopenev == 1 and openmanual==0){
    openev = 0;
    //Console.println("Grifo abierto con todos los sensores ok");
  }else{
    openev = 1;
    //Console.println("Grifo cerrado con todos los sensores ko");
  }
}
```

Enviar dades al servidor web

Amb aquesta funció enviem les dades al servidor web. Es una funció parametrizada per poder utilitzar la mateixa als diferents sensors.

```
int conexionservidor(int servsensor, int servvalor){
    delay(2500);
    Console.println("connecting...");

    if (client.connect(server, 80))
    {
        Console.print("connected");
        //Console.println(servsensor);
        //delay(2500);
        //hum = 55;

        switch (servsensor){
            case 1:
                //temperatura
                parametri="sensor=1&valor="+String(servvalor);
                break;
            case 2:
                //llum
                parametri="sensor=2&valor="+String(servvalor);
                break;
            case 3:
                //terra
                parametri="sensor=3&valor="+String(servvalor);
                break;
            case 4:
                //grifo
                parametri="sensor=4&valor="+String(servvalor);
                break;
            default:
                parametri="sensor=0&valor=0";
                break;
        }

        //Console.println(parametri);

        //http://localhost/RegDuino/ajaxfunciones.php?valor=26
        client.println("POST /ajaxfunciones.php HTTP/1.1");
        client.print("Content-length:");
        client.println(parametri.length());
    }
}
```

```
Console.println(parametri.length());
Console.println(parametri);
client.println("Connection: Close");
client.println("Host:regduino.javims.es");
client.println("Content-Type: application/x-www-form-urlencoded");
client.println();
client.println(parametri); //Set Remote temperture to ____
//client.print(temp);
//client.println("{}");
}
else
{
Console.println("connection failed");
}

return 0;
}
```


Impediments del projecte

Impediment 1:

La impressió de l'engranatge amb la impressora 3d, per culpa d'un disseny mal fet a retrasat el projecte, ja que es va imprimir amb les mesures més petites i no hi havia forma de poder tornar-la a imprimir. Això a fet que es retrases el muntatge del motor amb l'aixeta.

Impediment 2

El servomotor no te suficient força per a obrir l'aixeta. Encara no s'ha trobat una solució a aquest problema.

Impediment 3

Per falta de temps i diferents impediments al plantjament inicial del projecte, s'han hagut de retallar tasques a fer com per exemple, l'autosuministrament d'energia electrica, a través d'una placa solar.

Annex 1

S'adjunta com a document extern "tresoreria.pdf"

Referencias

REST API acces PHP

<https://www.techiediaries.com/php-jwt-authentication-tutorial/>

Informació relacionada amb les plaques solars.

<https://www.sfe-solar.com/paneles-solares/calcular-paneles-solares-necesarios/>

Get time Arduino yun

<https://www.arduino.cc/en/Tutorial/TimeCheck>

Control LED via web Arduino

<https://startingelectronics.org/tutorials/arduino/ethernet-shield-web-server-tutorial/web-server-LED-control/>

Insertar valores a bbdd desde arduino

<https://jadasupport.wordpress.com/2015/08/28/conectar-arduino-a-base-de-datos-mysql/>

Placa solar:

- Explicación detalle código:
<http://forum.arduino.cc/index.php?topic=465346.0>
- Modelo soporte placa solar para impresora 3d:
<https://www.thingiverse.com/thing:2939509>
- <https://www.thingiverse.com/thing:451728>

Agua:

- Modelo tubo para garrafa
<https://www.thingiverse.com/thing:428168>

Sensor suelo:

- <https://www.luisllamas.es/arduino-humedad-suelo-fc-28/>

Engranaje grifo:

<https://www.thingiverse.com/thing:1065755/comments>

<https://www.thingiverse.com/thing:1294905>

<https://www.thingiverse.com/thing:1938710>

<https://www.thingiverse.com/thing:3003792>

Estudio competencia.

<https://www.arielmax.com/proyecto-arduino-riego-automatico-para-plantas/>

Coeficiente de cultivo

<https://www.portalfruticola.com/noticias/2016/08/14/que-es-le-coeficiente-de-cultivo-kc-en-riego-valores-por-especie/>

Caudal agua

https://es.wikipedia.org/wiki/C%C3%A1lculo_del_caudal_de_agua_en_tuber%C3%ADas

Calculo de necesidades de agua

<http://elriego.com/informacion-tecnica/calculos/calculos-agua-tiempos-riego/calculo-necesidades-diarias-agua/>

Recoger Fecha i hora

<http://panamahitek.com/obtener-la-fecha-y-hora-con-el-arduino-yun/>

Repositorio models per a Fritzing

<https://github.com/adafruit/Fritzing-Library/tree/master/parts>

Agraiments

En col·laboració amb:

Per a crear les peces amb impressora 3d necessaries per al projecte s'ha necessitat l'ajuda de W!LAB de viladecans.

Concretament s'han impress:

- L'engranatge de l'aixeta.
- El minitub de la galleda d'aigua
- L'estructura de la placa solar.

<http://www.viladecans.cat/es/wlab>