



Desenvolupament d'un sistema de reutilització d'aigües grises basat en Arduino

Javier Gonzalez Fernandez
Grau Multimèdia
Arduino

Oriol Jaumandreu Sellarès
Pere Tuset Peiró

15/01/2017



Aquesta obra està subjecta a una llicència de
Reconeixement-NoComercial-SenseObraDerivada 3.0
Espanya de Creative Commons

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Desenvolupament d'un sistema de reutilització d'aigües grises basat en Arduino</i>
Nom de l'autor:	<i>Javier Gonzalez Fernandez</i>
Nom del consultor/a:	Oriol Jaumandreu Sellarès
Nom del PRA:	Pere Tuset Peiró
Data de lliurament (mm/aaaa):	<i>01/2017</i>
Titulació o programa:	<i>Grau Multimèdia</i>
Àrea del Treball Final:	<i>Arduino</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Aigües, grises, arduino</i>
<p>Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i></p>	
<p>Aquest treball final de grau té la finalitat de presentar un producte basat en Arduino, el qual tingui l'habilitat de recuperar l'aigua utilitzada en el rentamans (aigües grises) i enviar-la a la cisterna del inodor per ser reutilitzada.</p> <p>A través d'Arduino s'interconnecta i es dissenya el codi necessari de cadascun dels dispositius requerits per recuperar l'aigua.</p> <p>A més el projecte incorpora una aplicació web on es mostra amb detall el número de litres recuperats. On es poden analitzar els resultats del funcionament del dispositiu.</p> <p>Una vegada realitzat el desenvolupament i la posterior instal·lació del dispositiu, així com del servidor web, s'ha analitzat els resultats i aquests mostren que anualment es podria recupera 5.807 litres d'aigua i que la inversió es podria recuperar, depenent del tipus d'instal·lació, en 9 anys.</p> <p>Després de dur a terme el mencionat estudi i els mencionats resultats, s'arriba a la conclusió que el dispositiu ajuda a recuperar aigua i estalviar, però que en certes instal·lacions faria falta la implementació de tercer elements que permetin ser més eficient a la cisterna i, així, reduir el número de litres que es rebutgen quan es produeix la descarrega d'aigua.</p>	

Abstract (in English, 250 words or less):

This Bachelor degree final project has the purpose of release a product develops over Arduino, which has the ability to retrieve the water used from the basin (grey water) and send it to the toilet cistern, where it will be reused.

The devices are interconnected through Arduino and writing the needed code to retrieve the water.

In addition, an application web has been developed for this project, which displays the detail of the litres retrieved by the system.

Once the development is over and the device is installed along with the web server, the results have been analysed and this shows that, annually, it could retrieve 5.807 litres of water and that the inversion could be recovered, dependent of installation type, in 9 years.

Finally, with the mentioned analysis and the post-results, the conclusion is that the device can retrieve water and save Money. However, for some toilets, will be required to fit third party elements or devices, which permits the cistern to be more efficient. And then, reduce the number of litters which are thrown away when toilet flusher is activated.

Índex

1. Introducció	9
1.1. Títol del projecte	9
1.2. Context i justificació del Treball.....	9
1.3. Hipòtesi inicial.....	10
1.4. Objectius del Treball	11
1.5. Enfocament i mètode seguit	12
1.6. Planificació del Treball	13
1.7. Pressupost.....	16
1.8. Breu resumari de productes obtinguts.....	17
1.9. Breu descripció dels altres capítols de la memòria	17
2. Desenvolupament	18
2.1. Microcontrolador	18
2.1.1. Arduino	19
2.1.2. Selecció del controlador.....	22
2.1.3. Arduino Micro Pro	23
2.1.4. NodeMCU v3	29
2.2. Vàlvula d'entrada.....	30
2.3. Bomba d'aigua.....	32
2.4. Vàlvula de tres vies.....	34
2.5. Sensor de nivell	36
2.6. Tanc d'aigua	38
2.7. Sensor de flux d'aigua	40
2.8. Assemblatge del Prototip	42
2.9. Producte final.....	43
3. Milliores	45
3.1. Integració del mòdul NodeMCU	45
3.2. Enviament de dades	46
3.3. La interfície d'usuari.....	49
4. Resultats	54
5. Conclusions	57
6. Glossari	60
7. Bibliografia	64
8. Annexos	66
Annex I: Codi dels dispositius	66
Annex II: Explicació desenvolupament interfície d'usuari.....	71
Annex III: Instal·lació Servidor Web per realització	74
Annex IV: Creació de la base de dades	80
Annex V: Implementació de l'aplicació web.	82
Annex VI: Valors obtinguts durant el període de prova.	84

Índex de figures

Figura 1. Diagrama GANTT - planificació del projecte	15
Figura 2. Arduino Micro Pro	23
Figura 3. Diagrama de pins de la placa Micro Pro.....	25
Figura 4. ESP8266-01	26
Figura 5. Esquema de connexions ESP- 01.....	27
Figura 6. Diagrama de pin de la placa NodeMCU v3	30
Figura 7. Esquema de connexions de la vàlvula d'entrada.....	31
Figura 8. Vàlvula magnètica.....	32
Figura 9. Esquema de connexions de la bomba d'aigua	33
Figura 10. Bomba d'aigua	33
Figura 11. Esquema de connexions de la vàlvula de tres vies	35
Figura 12. Esquema de connexions del sensor de nivell.....	37
Figura 13. Tanc d'aigua	38
Figura 14. Sensor de flux d'aigua.....	40
Figura 15. Esquema de connexions del sensor de flux d'aigua.....	41
Figura 16. Prototip inicial	42
Figura 17. Prototip inicial - Sensor de flux.....	42
Figura 18. Tanc d'aigua instal·lat	43
Figura 19. Bomba d'aigua instal·lada	43
Figura 20. Sensor de flux i fontaneria	43
Figura 21. Diagrama de connexions generals.....	44
Figura 22. Diagrama de connexions amb el mòdul NodeMCU	45
Figura 23. Definició llibreria ESP8266Wifi.h.....	45
Figura 24. Definició dels detalls de la connexió WIFI	46
Figura 25. Codi de connexió WIFI.....	46
Figura 26. Creació de la taula de registres.....	47
Figura 27. Codi de connexió amb MySQL.....	47
Figura 28. Arduino connectant amb el portal.....	48
Figura 29. Codi de creació de la URL	49
Figura 30. Sumari de litres estalviats	50
Figura 31. Gràfica de litres recuperats mensualment.....	51
Figura 32. Gràfica de litres recuperats diàriament.....	51
Figura 33. Aplicació Web	52
Figura 34. Accés a l'aplicació Web	53
Figura 35. Gràfica de litres estalviats diàriament.....	54
Figura 36. Definició llibreria de comunicacions sèrie.....	66
Figura 37. Paràmetres de comunicació sèrie amb ESP8266-01	66
Figura 38. Codi de comunicació sèrie amb el ESP-01.	67
Figura 39. Llibreria de la vàlvula d'entrada.....	67
Figura 40. Llibreria de la Bomba d'aigua.....	68
Figura 41. Codi verificació del estatus pin 2.....	69
Figura 42. Configuració dels <i>interrups</i> en la funció <i>setup()</i>	69
Figura 43. Definició de la funció comptadora de polsos.	69
Figura 44. Funció del càlcul de litres basats en els polsos detectats.....	70
Figura 45. Definició de <i>canvas</i> en HTML.	71
Figura 46. Definició dels scripts en HTML.....	71
Figura 47. Trucades AJAX a sentències PHP.....	71
Figura 48. Codi PHP de connexió i consulta amb la BD per la posterior impressió en JSON.....	72
Figura 49. Configuració dels gràfics en script <i>chart.js</i>	73

Figura 50. Selecció de idioma en el instal·lador de wamp.....	74
Figura 51. Acceptació dels termes i condicions.....	75
Figura 52. Nota informativa.....	75
Figura 53. Carpeta de destí de la instal·lació	76
Figura 54. Accés directe en el menú d'inici.	76
Figura 55. Sumari de instal·lació.....	77
Figura 56. Procés de instal·lació.....	77
Figura 57. Selecció navegador web.....	78
Figura 58. Nota informativa final de instal·lació.....	78
Figura 59. Finestra de instal·lació completa.....	79
Figura 60. Barra de tasques amb wamp iniciat.	79
Figura 61. Finestra de inici phpMyAdmin.	80
Figura 62. Finestra de importació phpMyAdmin.	81
Figura 63. Botó Continuar en la finestra de importació phpMyAdmin.	81
Figura 64. Carpeta arrel de wamp.....	82
Figura 65. Fitxer de l'aplicació web.	83

Índex de taules

Taula 1. Productes destacats al mercat.	10
Taula 2. Costos del dispositiu	16
Taula 3. Característiques Arduino MicroPro.....	24
Taula 4. Característiques del NodeMCU v3	30
Taula 5. Resultats número de litres estalviats diàriament	55
Taula 6. Estalvi en cada moment de la prova.	85

1. Introducció

1.1. Títol del projecte

Les aigües grises són aquelles que surten pels desguassos de les banyeres, dels rentamans, de les piles de la cuina, de la rentavaixella o de les rentadores. Aquestes aigües, amb un tractament senzill, poden ser fàcilment reutilitzades per a diversitat d'usos. El més comú d'aquests usos és emprar-la per emplenar les cisternes dels vàters que no requereixen aigua de gran qualitat. Però, també, es poden emprar per a altres usos, com el reg de zones verdes o la neteja d'exterior.

El desenvolupament d'un sistema de reciclatge d'aigües grises, basat en Arduino, és el títol escollit pel projecte, el qual es vol posar l'atenció sobre la recuperació de les aigües rebutjades del rentamans.

Aquest nom resumeix en un titular la posterior funcionalitat que aquest dispositiu tindrà, tal com veurà a continuació. Així com, el què es vol aconseguir i el mitjà pel qual s'aconseguirà.

1.2. Context i justificació del Treball

Segons les últimes dades, gairebé, dos milions de persones es moren a l'any per falta d'aigua potable^[1]. Hi ha més de mil milions de bilions de litres d'H₂O en el nostre planeta, però més del 97% de l'aigua a la Terra és salada. Això deixa disponible per a la majoria dels éssers vius una fracció mínima.

A més, des de 1950, la població mundial s'ha duplicat i en l'actualitat el nombre d'habitants ronda els 7.000 milions. I aquest valor no para de créixer.

Aquesta situació d'escassetat que es viu dia a dia, ja no tan sols es vigent en països que no són el nostre, sinó que ja en el nostre propi es veuen restriccions d'accés a l'aigua.

Donat el nivell d'aigua gastada o, es podria dir, malgastada per la humanitat, es de rebut plantejar-se si es podria verificar la possibilitat de trobar alternatives que ajudin a esmenar i/o solucionar, en gran mesura, aquest gran problema.

Segons una primera recerca, el mercat ofereix alternatives les quals no es poden utilitzar a petita escala i tampoc semblen ser atractives o assequibles pels usuaris. Donat que els barems de preus solen oscil·lar, aproximadament, entre 300 € – 5000 €. Això dependrà de les tecnologies i les funcionalitats que es requereixin.

A continuació es pot veure una gràfica amb els dispositius més destacats al mercat encarregats de depurar i/o recuperar l'aigua rebutjada en l'àmbit domèstic.

Marca	Model	Característiques principals	Preu
Sidar	AT6 ^[2]	- Depura Aigües residuals - Instal·lació Subterrània	3.187 €
Sidar	ATC-P 12 ^[3]	- Depura Aigües residuals - Gran capacitat - Instal·lació Subterrània	5413 €
Ecohoe	AQUS ^[4]	- Fàcil instal·lació.	786,50 €
Basement Pumping	600 Series ^[5]	- Gran capacitat (150 l). - Instal·lació fora del lavabo.	£373.68
Aquawise	GW210 ^[6]	- Instal·lació Subterrània	£1995

Taula 1. Productes destacats al mercat.

1.3. Hipòtesi inicial

Amb les dades anteriors, si n'hi ha alternatives, són aquestes o aquesta l'alternativa viable per la societat i pel usuari final?

En el cas de que sigui viable, seria possible la recuperació de la inversió en un període de temps relativament curt, com per exemple 2 a 5 anys?

I la més important, seria possible la recuperació d'aigua suficient, com per impulsar aquestes alternatives o solucions?

Davant aquestes qüestions, el plantejament inicial seria la recerca d'accions o coses quotidianes, per les quals el màxim nombre d'usuaris possible vegi viable la incorporació de noves accions per tal de malgastar el mínim possible.

En aquest punt, per exemple, s'observa que els agricultors emmagatzemen i utilitzen l'aigua de la pluja pel regadiu. Per tant, no es podria recuperar l'aigua que rebutgen de casa per altres funcions? No es podria utilitzar l'aigua que rebutgen del rentamans, dutxes, rentavaixelles, rentadora, entre d'altres, per la carrega de la cisterna del wàter?

Depenent de la seva eficiència de la cisterna, cada vegada que es produeix una descarrega, es rebutgen entre 3 i 10 litres d'aigua. I tenint en compte que, de manera normal, el número de vegades que una persona hauria de passar pel inodor són de 5 a 8 vegades. Cada persona està rebutjant entre 15 i 80 litres d'aigua diàriament.

Això suposa que cada família estàndard de 4 membres està rebutjant cada any, de mínim, una mitjana de 21.900 litres i un màxim 116.800 litres. Així es pot conèixer que

per família, tenint en compte que el preu de l'aigua es pròxim al 2 euros per metre cúbic, el inodor suposa un cost de 42€ a 233€ euros cada any.

Segons l'INE^[7], en 2009, els espanyols consumien de mitjana 149 litres per habitant i dia, dels quals el 18% s'utilitza en el rentamans. Amb aquestes dades, s'observa que cada família rebutja des del rentamans cap al clavegueram de mitja 107 litres diàriament.

Depenent de la instal·lació, es considera que es poden recuperar entre 30% i 50% del total rebutjat pel rentamans. Així doncs, cada família podria estalviar entre 32 i 54 litres diàriament, és a dir, entre 11.800 i 19.710 litres anualment. Dit d'una altra manera, l'equivalent a un estalvi monetari d'entre 23.6€ i 39.42€ anual.

En aquest punt, s'hauria de revisar el condicionant referent al volum d'aigua que cada família pot emmagatzemar. Per exemple, en un tanc d'aigua 25 litres, si no es produeixen descarregues, és a dir, si no es produeix un cicle d'aigua, el tanc arribarà al seu nivell màxim i començarà a rebutjar l'excés d'aigua que es genera dels mencionats 32 i 54 litres.

L'oportunitat que ens brinda la tecnologia, com la que aporta Arduino, posa a l'abast de molts una eina fàcil, assequible i versàtil. Amb aquesta idea es pot veure si a menor escala es poden aconseguir gran resultats.

1.4. Objectius del Treball

Els objectius del projecte es poden categoritzar en dos conceptes. Aquests van lligades entre si per tal d'aconseguir el producte final.

Primerament, l'objectiu principal, serà:

- La presentació d'un producte amb la capacitat de recuperar les anomenades aigües grises que es rebutgen des del rentamans.
- Analitzar els resultats obtinguts per tal de saber la viabilitat del projecte, així com en quin moment es pot recuperar la inversió inicial, tenint en compte els costos i les mencionades variables e hipòtesis.

De manera secundària, i lligada amb el primer objectiu, el següent serà la demostració de què es viable la instal·lació d'aquests dispositius, tant per l'estalvi d'aigua com el posterior estalvi econòmic.

Tenint en compte aquest dos objectius generals, es considera com objectius específics:

- Idear i desenvolupar les especificacions que es requereixen pel producte.
- Fer operatius cadascun dels elements que componen el producte final
- Elaborar les proves i posteriors correccions, amb la finalitat de que es compleixin les expectatives del objectiu principal.
- Implantar el producte per que pugui ser utilitzat pel usuari final.
- Analitzar la informació recopilada del producte final i organitzar-la per les posterior conclusions.

1.5. Enfocament i mètode seguit

A partir d'aquí s'ha de dissenyar una estratègia d'actuació, a qui es vol dirigir el producte, i com es vol dirigir.

1.5.1. Públic objectiu.

Aquest projecte no solament està dirigit a aquells *makers* que realitzen proves i projectes basats en la tecnologia d'Arduino. La idea intenta posar a l'abast de qualsevol persona, empresa i organisme, un sistema barat i fàcil de instal·lar. Manejable per usuaris i instal·ladors sense necessitat d'un nivell tècnic avançat.

Aquests tècnics o usuaris podran instal·lar aquests dispositius al costat de qualsevol vàter en zones residencials, industrials i, fins i tot, públiques.

1.5.2. El projecte i el mercat.

Com ja s'ha comentat, els principals avantatges i beneficis que aquest producte pot aportar respecte del mercat, són que es posa a l'abast del mercat un dispositiu relativament més barat, fàcil de produir i de gran qualitat. Aportant aquestes eines, aconseguim que es redueixin, en gran manera, els nivells de consum d'aigua, els quals dia a dia va en augment.

Atès que és una eina molt útil i donats els nivells crítics, és considerat que aquesta ha de ser compartida amb qualsevol usuari disposat ajudar al planeta. Posant a la disposició d'aquests i facilitant les eines, tecnologia i coneixements que siguin necessàries per a l'engegada d'aquests dispositius.

I així, intentar que el mercat pugui adaptar-se a les circumstàncies abans esmentades. Atès que en aquest no es faciliten a un preu raonable les tecnologies necessàries per a la implantació local.

És cert que els preus han estat reduïts al llarg dels últims anys. Donat, també, que els elements tecnològics i els preus han estat reduïts. No obstant això, els costos, nivells de distribució i promoció no han estat prou assequibles perquè els usuaris considerin a aquestes suficientment rentables.

Per aquests motius, la millor estratègia és desenvolupar:

1. Un producte el més assequible possible econòmicament.
2. Un producte que sigui el més simple possible dintre de les possibilitats
3. Fàcil de manipular e integrar per la comunitat.

Considerem que, actualment, i donada la situació, tant econòmica com social, la millor estratègia per aconseguir, posteriorment, la divulgació d'aquest es garantir que aquest tipus d'eines siguin accessible per qualsevol públic.

1.6. Planificació del Treball

Una vegada definides les bases i el per què d'aquest projecte, s'han de distingir quines són les àrees a treballar, els recursos que es necessiten, com es plantejarà el prototip i quins elements es requeriran. Tot això per tal d'elaborar i plantejar les tasques i àrees que necessiten completar d'aquest projecte:

Dispositiu

- Definició del requeriments del disseny de la solució
- Recerca i recopilació de la informació
 - Recerca dels elements del dispositiu final
 - Informació sobre el seu funcionament
- Prototip
 - Creació de les funcions necessàries pel control diferents dispositius
 - Assemblatge dels elements del prototip i el seu codi
- Unificació del codi i assemblatge del producte
- Test del dispositiu
- Rectificacions i millores

- Validació i prova final
- Implantació

Documentació

- Memòria

Presentació.

- Recerca de recursos audiovisuals
- Disseny gràfic i visual de l'exposició final
- Creació del recursos audiovisuals per l'exposició final
- Divulgació dels recursos audiovisuals

Així doncs, una vegada es té definit els recursos què es faran ús i es té identificat cadascuna de les tasques a realitzar, es plasma de manera gràfica cadascuna d'elles amb els temps que es trigarà per la seva execució.

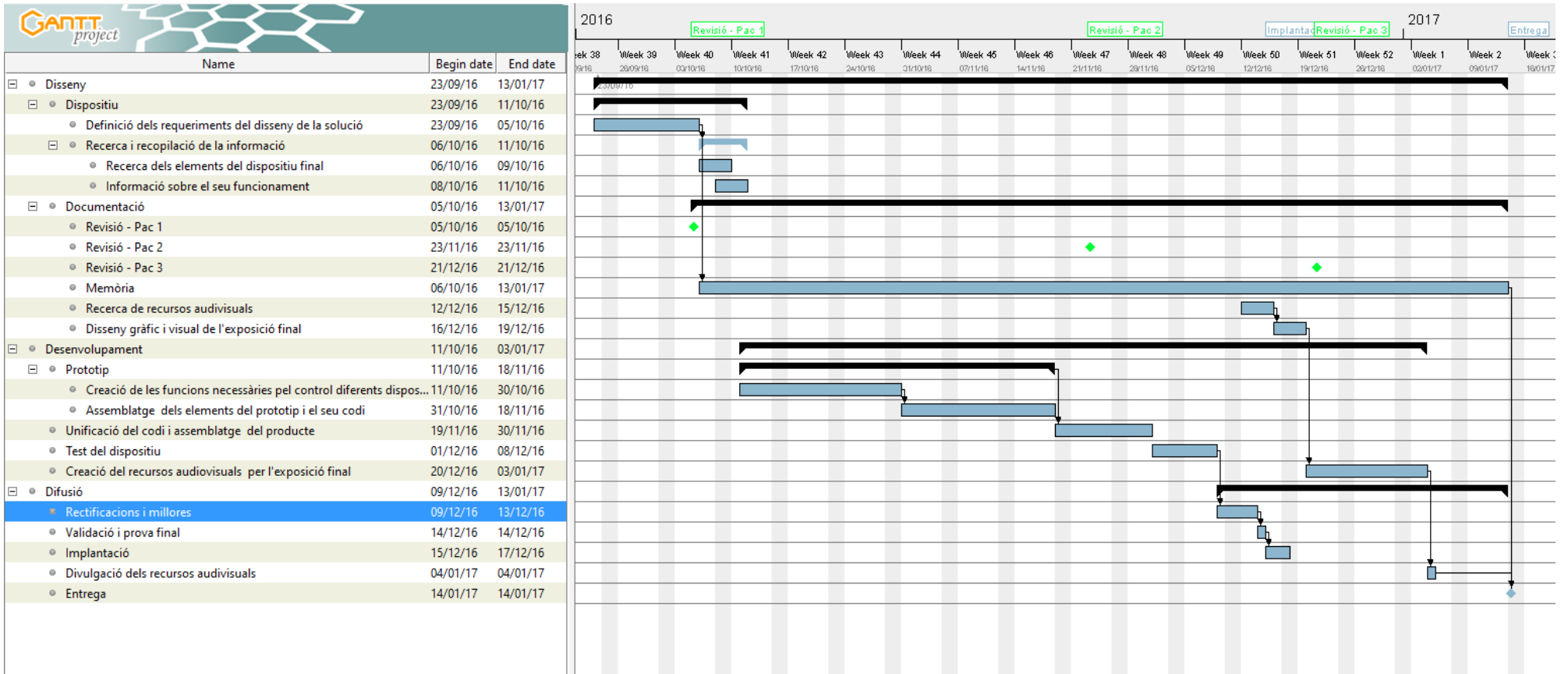


Figura 1. Diagrama GANTT - planificació del projecte

1.7. Pressupost

A continuació es defineix el pressupost necessari per a la realització del projecte, sobre la base dels recursos necessaris, tant humans com a materials, per donar-lo a terme, i d'acord a la planificació plantejada.

De forma general, en relació als recursos dels materials necessaris, el pressupost es pot resumir de la següent forma:

Element		Cost per unitat	Emprats en el dispositiu final	
Software	Arduino IDE	0 €	<input checked="" type="checkbox"/>	
	Mysql (Base de dades)	0 €	<input checked="" type="checkbox"/>	
	Elements Servidor Web	0 €	<input checked="" type="checkbox"/>	
Hardware	Microcontrolador + Antena Wifi	Arduino Micro Pro	2.91 €	<input type="checkbox"/>
		ESP-01	1.45 €	<input type="checkbox"/>
		Node MCU V3	2.63 €	<input checked="" type="checkbox"/>
	Sensor de flotació	0.89 €	<input checked="" type="checkbox"/>	
	Vàlvula de 3 vies	23.95 €	<input checked="" type="checkbox"/>	
	Vàlvula magnètica	6.65 €	<input type="checkbox"/>	
	Bomba d'aigua	11.85 €	<input checked="" type="checkbox"/>	
	Sensor de flux	8.06 €	<input checked="" type="checkbox"/>	
	Depòsit d'aigua	40.87 €	<input checked="" type="checkbox"/>	
	Font d'alimentació 12V	8.50 €	<input checked="" type="checkbox"/>	
Altres elements (circuiteria, tuberies, etc.)	12 €	<input checked="" type="checkbox"/>		
Cost Total		119.76 €		
Cost dispositiu final		108.75 €		

Taula 2. Costos del dispositiu

1.8. Breu sumari de productes obtinguts

Més avall, en la fase de desenvolupament, es podrà veure amb més detall els productes obtinguts a partir del procediment realitzat.

Aquests s'han aconseguit a partir de la definició de tres fases on, primerament hem posat en funcionament els elements de manera individual. Aconseguint el funcionament de:

- Vàlvules: Les quals es podem utilitzar per aturar o obrir la circulació d'aigua per a la instal·lació de un sistema regadiu, per exemple.
- Bombes d'aigua: Aquestes també podem utilitzar-se per la instal·lació anterior. O el funcionament d'un aquari. Així com, pel bombeig d'aigua en piscines. Aquest últim, comportaria la instal·lació d'una bomba més potent, però el codi desenvolupat es podria fer servir.
- Comptador d'aigua. Els quals de manera individual poden ser utilitzats per altres menesters com d'instal·lació a la entrada d'aigua d'una casa o empresa i portar el control d'aigua empleat.

Segonament, hem vist com funcionaven els dispositius de manera conjunta. Aquesta part ens ha ajudat a establir quins altres elements ens feien falta. Però aquest producte ja era totalment funcional.

I per últim el producte final amb tots els dispositius connectats i treballant de manera conjunta.

1.9. Breu descripció dels altres capítols de la memòria

A continuació, s'especifiquen tres grans blocs:

1. Desenvolupament
2. Millores
3. Resultats.

L'apartat de desenvolupament incorpora de manera individual i col·lectiu de cadascun dels elements que componen el dispositiu. Així com els diferents plantejaments i alternatives que es poden trobar al mercat.

La secció de millores detalla cadascuna de les noves implantacions sobre el dispositiu final. Així com el per què d'aquests nous desenvolupaments.

Finalment, els resultats mostren el detall de les dades obtingudes de la utilització del prototip una vegada en funcionament.

2. Desenvolupament

Com s'ha mencionat, anteriorment, es pretén desenvolupar un sistema que tingui la facultat de recuperar l'aigua que es rebutja del rentamans, tot ell, gestionat a partir de un microcontrolador com és Arduino.

Dintre de la fase de desenvolupament s'han definit tres grans blocs o procediments establerts per arribar al producte final. Aquestes han estat:

- a. Primera fase de prototipatge
- b. Segona fase de prototipatge
- c. Producte final

La primera fase comporta la posta en funcionament de cadascun dels dispositius de manera individual. És a dir, fent ús l'eina de programació d'Arduino s'ha desenvolupat el codi principal el qual pot posar funcionament cadascun dels dispositius.

La segona fase consisteix en la creació del prototip que ens ajudarà a tenir una idea de com quedaria el producte final. Durant aquesta fase, s'han assemblet els elements i s'ha implementat el codi de cadascun dels dispositius en un únic producte.

A continuació, es defineixen cadascun dels elements empleats en les fases de prototipatge, com són la posta de funcionament de manera separada de cadascun dels elements o dispositius que componen el projecte, el per què de la seva elecció i alternatives, així com el seu funcionament.

2.1. Microcontrolador

Un microcontrolador és un circuit integrat digital programable, que pot executar ordres i funcions prèviament gravades en memòria.

Aquest pot utilitzar-se per a moltes aplicacions, algunes d'elles són: maneig de sensors, controladors, jocs, calculadores, avisos lumínics, seqüenciador de llums, control de motors, rellotges, alarmes, robots, entre unes altres.

Donat que el *hardware* ja ve integrat en un sol xip, per utilitzar un microcontrolador s'ha d'especificar el seu funcionament per programari a través de programes que indiquin les instruccions que el microcontrolador ha de realitzar. En una memòria es guarden els programes i un element anomenat CPU s'encarrega de processar pas per pas les instruccions del programa. Els llenguatges de programació típics que s'usen per a aquesta fi són assemblellador i C, però abans de gravar un programa al

microcontrolador cal compilar-ho a hexadecimal que és el format amb el qual funciona el microcontrolador.

Per dissenyar programes és necessari conèixer els blocs funcionals bàsics del microcontrolador, aquests blocs són:

- CPU (Unitat central de procés)
- Memòria ROM (Memòria de solament lectura)
- Memòria RAM (Memòria d'accés aleatori)
- Línies d'entrada i sortida (Perifèrics)

La forma en la qual interactuen aquests blocs dependrà de la seva arquitectura

2.1.1. Arduino

Arduino és una plataforma de codi obert, basada en una placa amb un microcontrolador (*hardware*) i un entorn de desenvolupament (*software*), dissenyada per facilitar l'ús de l'electrònica en projectes multidisciplinaris.

Existeix en el mercat una àmplia gamma de plaques Arduino oficials. Totes elles, inclouen com a element principal microcontroladors de la família tecnològica AVR (de la marca Atmel), com el model Uno, Mega o Nano que inclouen chips com el ATmega328, ATmega32u4, el ATmega2560, entre d'altres.

El **hardware** és el conjunt de components que integren la part material de Arduino, és a dir, la pròpia placa de Arduino. Es tracte d'una targeta programable, en forma de circuit imprès (*Printed Circuit Board*) en el qual la superfície està fabricada d'un material no conductor i traçada amb pistes d'un material conductor que connecten elèctricament els diversos components soldats a la placa.

L'element principal és el microcontrolador, en el nostre cas és de 8 bits de resolució que consta d'una memòria EEPROM (*Electrically Erasable Programmable Read-Only Memory*) de 512 bytes/1KB. El microcontrolador té una sèrie de connectors o pins que s'utilitzen com entrada i sortida, registres de memòria, un oscil·lador intern que facilita al microcontrolador una sèrie de polsos per mantenir estable la velocitat del treball, e interfases per la comunicació sèrie; entre els seus principals components.

Els pins del microcontrolador, es troben units internament als pins de la placa. Així, a través de la placa i del IDE Arduino, mitjançant sensors es pot extreure informació del món físic, és a dir, de l'entorn que ens envolta. A més, mitjançant actuadors o activadors, com leds, altaveus, displays, motors o pantalles, es pot proporcionar informació. Per això, les sortides o “pins de sortida” de Arduino ofereixen valors elèctrics.

Alguns models de plaques Arduino tenen sensors i actuadors integrats però, en general, es treballa amb sensors i actuadors externs, i amb una placa de prototips que simula el futur circuit imprès, i en la qual es poden muntar i desmuntar fàcilment les connexions mitjançant cables. A més, als diferents models de plaques Arduino, se'ls poden afegir *shields* que són plaques complementàries que mantenen la mateixa disposició que els pins de les plaques Arduino, per apilar-los còmodament sobre elles, simplificant connexions més complicades.

Les plaques es poden construir-se a mà o comprar-se ja preensamblades i el IDE oficial per treballar amb Arduino es pot descarregar-se de forma gratuïta. A més, els desenvolupaments amb Arduino poden requerir d'un sistema operatiu per executar-se, o no (*stand-alone*), o fins i tot interactuar amb un programa que s'està executant en un ordinador com és el flash, per exemple.

A més, la placa Arduino consta d'una sèrie d'entrades i sortides en forma de connectors o pins tant analògiques com digitals. Per una banda, els pins d'entrada o sortida digital es poden comportar-se com el seu propi nom indica segons se especifica en el *sketch* programat en el IDE. Per una altra banda, les entrades analògiques accepten valors analògiques, és a dir, lectures de voltatge des d'un sensor i les converteixen en un valor comprès entre 0 i 1023; i les sortides analògiques són sortides digitals reprogramades per oferir valors de sortida analògiques mitjançant el *sketch* creat en el IDE.

La placa, per funcionar, necessita alimentació elèctrica que pot provenir d'una font d'alimentació externa o bé des del ordinador al que estigui connectada la placa mitjançant un cable USB. Si no hi ha una font d'alimentació externa, l'alimentació vendrà pel USB però en quan s'endolli una font d'alimentació, la placa la utilitzarà automàticament.

El disseny original d'Arduino inclou els components electrònics per alimentar i comunicar-se amb el microcontrolador així que altres alternatives de disseny haurien de incorporar-se o al menys, mostrar la forma d'afegir-los fàcilment.

En quan al *software*, per a programar el *hardware* d'Arduino es necessari un entorn de desenvolupament o IDE (*Integration Development Environment*) en forma d'un conjunt d'eines software per a desenvolupar i provar els programes denominats *sketches* en Arduino.

El IDE oficial de Arduino es basa en el llenguatge de programació de codi obert *Processing*, dissenyat especialment per persones sense experiència en programació; i en la plataforma Wiring pel desenvolupament senzill de prototips d'aplicacions. Aquest IDE oficial, es pot descarregar-se des de la seva pròpia pàgina web de Arduino de forma gratuïta.

El llenguatge de programació de Arduino es coneix com *Arduino Programming Language*. Està basat en el llenguatge C, per la qual suporta qualsevol construcció estàndard C i algunes funcionalitats de C++. A grans trets, el llenguatge de programació d'Arduino, es pot dividir en tres grans grups: estructures o blocs (com *setup()*, *loop()*, *operadors*, etc.), valors variables i constants; i funcions. Existeixen a més, gran nombre de llibreries per a la interacció d'Arduino amb determinats tipus de *hardware* i nombroses contribucions de codi per part la comunitat.

El procés de compilació que realitza interiorment Arduino consisteix en la transformació del programa en llenguatge C++, creat a partir del codi en el llenguatge Arduino, en el programa en codi màquina (binari) AVR, amb l'extensió *.hex, executable pel microcontrolador.

En qualsevol procés de compilació se succeeixen en dos etapes. Una primera d'anàlisi per a dividir un programa font en els elements que el componen i per crear un programa intermedi; i una segona etapa de síntesi per a construir el programa objecte que serà transformat, mitjançant un *linker* en el programa executable.

El nucli de Arduino, el *toolchain* és la seqüència de programes que se invoquen cada vegada que se compila un projecte, generant un fitxer de codi màquina amb extensió *.hex que serà enviat a placa Arduino a través de la connexió USB o un programador ISP si no hagués el *bootloader* instal·lat. Durant

el procés de compilació de Arduino, se compilen tant el/ls sketch/es del projecte, les llibreries incloses, i els diferent fitxers que Arduino requereix.

2.1.2. Selecció del controlador

Una vegada seleccionada la plataforma de programació, és necessari triar la placa. A l'hora d'escollir, es van tenir en compte els següents requisits:

- **Mida:** Amb les necessitats mínimes, el controlador hauria d'ocupar el mínim possible. Donat que es vol ocultar el màxim possible i tenir més espais per altres dispositius.
- **Preu:** El cost de la placa és un dels factors que fan Arduino una eina versàtil i estesa en la comunitat. Així de la gama de plaques que tenim al mercat s'ha d'aconseguir tenir tot el necessari pel mínim cost.
- **Número de entrades y sortides:** Aquest no serà un problema, perquè gairebé totes les plaques d'Arduino presenten suficient entrades i sortides que s'adapten als requeriments mínims del projecte que es vol realitzar.
- **Interrupcions:** una interrupció és una senyal que interromp l'activitat normal del nostre microprocessador i salta a atendre-la. Hi ha tres esdeveniments que poden disparar una interrupció:
 - Un esdeveniment del hardware, prèviament definit.
 - Un esdeveniment programat.
 - Una trucada des del programari.

Quan un esdeveniment dispara una interrupció, l'execució normal del *micro* es suspèn i salta a executar una funció especial anomenada *Interrupt Service Handler* o *ISH* (Servei de gestió d'interrupció).

Quan el ISH finalitza, el processador torna tranquil·lament al punt on ho va deixar. A continuació, segueix amb el procés que estava realitzant com si no hagués passat gens.

En aquest cas es requereixen varis pins digitals amb aquesta funció integrada. Per tant, és un punt a tenir en compte perquè no tots els controladors de plaques Arduino incorporen el necessari per integrar tots els elements dels que es compona aquest projecte.

- **Consum:** Donat que els elements no requeriran estar contínuament en funcionament, es precisa d'una placa la qual pugui estar en funcionament amb el mínim consum possible.
- **Estandarditzat:** Un dispositiu que sigui conegut pel usuari, és a dir, que estigui estandarditzat en la comunitat. Per tant, sigui més fàcil i assequible de trobar exemples, ajuda e informació sobre el seu funcionament. A més, en estar estandarditzat, posteriorment, la implantació pot ser més assequible si el projecte requereix ser modificat o replicat per altres usuaris.

Amb les premisses anterior i l'estudi realitzat sobre el catàleg de plaques presentat per Arduino i els seus associats, el dispositiu que millor s'adapta als propòsits d'aquest projecte és "Arduino Micro Pro".

Aquest últim comentari ho posem entre cometes, perquè, com veurem a continuació, aquest dispositiu és perfecte pel propòsit inicial que és la de recuperar les aigües grises. Però, finalment, donat que com a millora es volia extreure les dades sense tenir que accedir el dispositiu, Arduino Micro requereix de mòduls externs per aconseguir aquest propòsit.

2.1.3. Arduino Micro Pro

Aquesta és una petita placa, la qual permet un disseny i una disposició dels diferents dispositius, que posteriorment es connectaran, gairebé igual de manejable que els seus germans més grans.

Aquest dispositiu va ser integrat directament per la companyia Sparkfun i integrat dintre del catàleg d'Arduino. Tot ell amb l'idea de proveir al mercat una eina potent, amb el mínim cost i una mida reduïda en comparació amb als altres elements del catàleg.

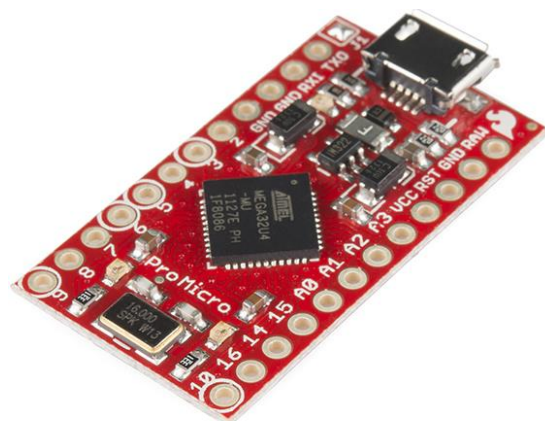


Figura 2. Arduino Micro Pro

Incorpora un microcontrolador ATmega32u4, el qual ve preprogramat amb un carregador d'arrencada que li permet carregar un nou codi sense l'ús d'un programador extern.

Un dels avantatges d'aquest nou microcontrolador és el que disposa d'USB natiu i , per tant, no necessita de cap conversió sèrie-USB, com solia ocórrer amb versions de mida reduïda.

Les seves característiques^[8] principals són:

Microcontrolador	ATmega32u4
Voltatge de funcionament	5V
Voltatge d'entrada recomanat	7-12V
Pins digitals I/O	12
Pins PWM Digital I/O	5
Pins analògics	4
Memòria flash	32 KB
SRAM	2.5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Freqüència del rellotge	16 MHz
Llargada	33.02 mm
Amplada	17.78 mm
Pes	Menys de 2 grans.

Taula 3. Característiques Arduino MicroPro

A continuació, es mostra el mapa o diagrama d'on està situats cadascun dels pins que incorpora la placa i les seves funcions.

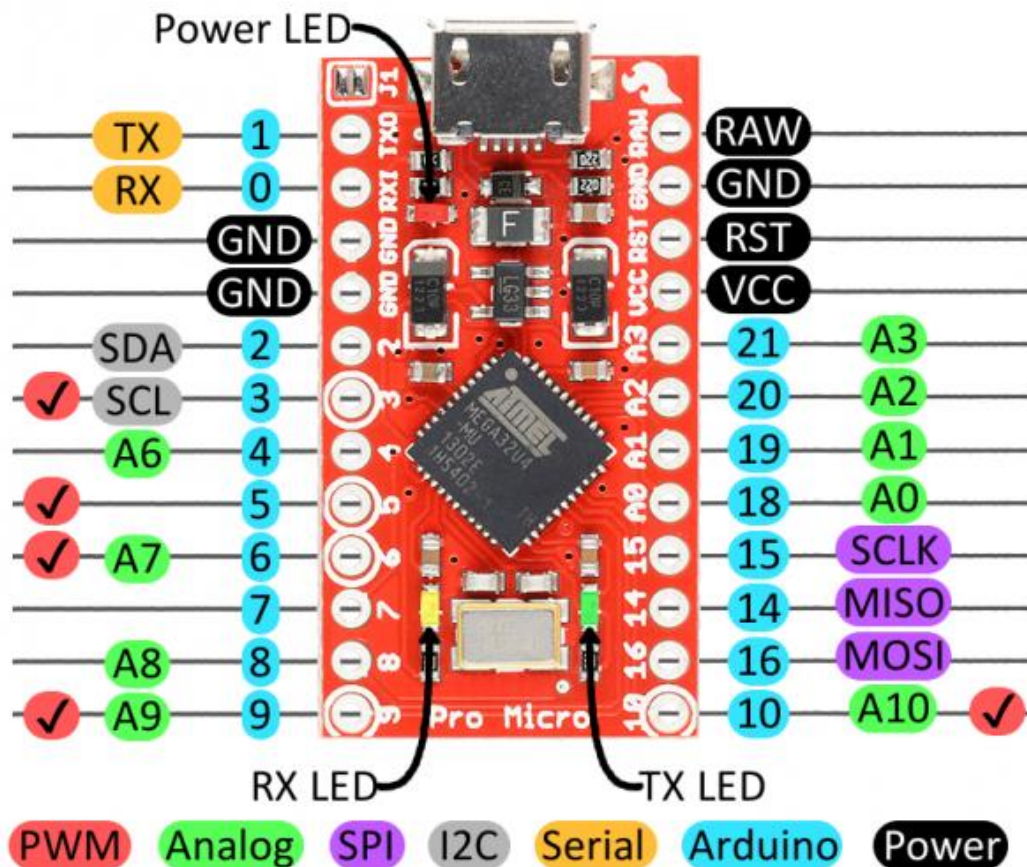


Figura 3. Diagrama de pins de la placa Micro Pro

Aquestes característiques, tant físiques com de processament, fan que el Pro Micro sigui la més apropiada. A més, donat que no es requereixen la majoria de les connexions, posteriorment, es podrien afegir nous components amb altres propòsits o millorar els que ja existeixen.

2.1.3.1 ESP8266-01

Aquest punt, s'ha definit, dintre d'Arduino Micro Pro, per tal de integrar-lo com una part del propi Micro Pro, encara que es un dispositiu extern. El propòsit de fer-ho d'aquesta manera es per mantenir o definir el mateix esquema del microcontrolador que finalment s'ha utilitzat.

El ESP8266 és un xip altament integrat dissenyat per a les necessitats d'un nou món connectat. Ofereix una solució completa i autònoma de xarxes WIFI, la qual cosa li permet allotjar l'aplicació o servir com a enllaç entre internet i un microcontrolador.

Aquests mòduls vénen per defecte integrats amb un firmware mitjançant el qual, a través dels pins RX i TX, es poden enviar comandes AT. Amb el propòsit de, per exemple, escanejar xarxes WIFI, fer peticions HTTP, utilitzar el mòdul com a client per connectar-ho a un *router*, convertir el mòdul en un AP, entre d'altres.

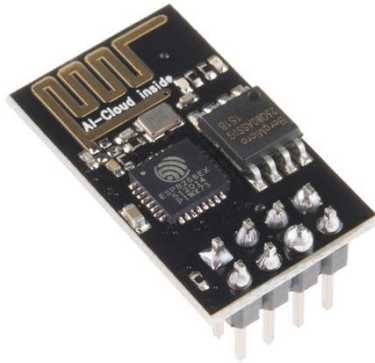


Figura 4. ESP8266-01

Hi ha al voltant d'una dotzena de diferents mòduls que implementen l'integrat ESP8266 amb diferents formats, amb diferents tipus d'antena o sense ella i amb més o menys pins d'entrada sortida disponibles. Entre tots d'ells els més difosos són el Esp-01 i el Esp-02.

2.1.3.1.1 Esquema de connexions

A part de l'alimentació, la qual ha de ser 3,3V, serà necessari connectar l'entrada i sortida sèrie URXD i UTXD a les corresponents del microcontrolador i activar l'integrat connectant el pin CHPD. El mòdul podria funcionar deixant sense connexió el pin RST (*reset*) però, per evitar que qualsevol alteració electromagnètica pogués reiniciar-lo, el més recomanable és connectar també aquest pin.

L'esquema següent mostra els elements utilitzats, connexions i circuit dissenyat per aconseguir el funcionament del mòdul WIFI.

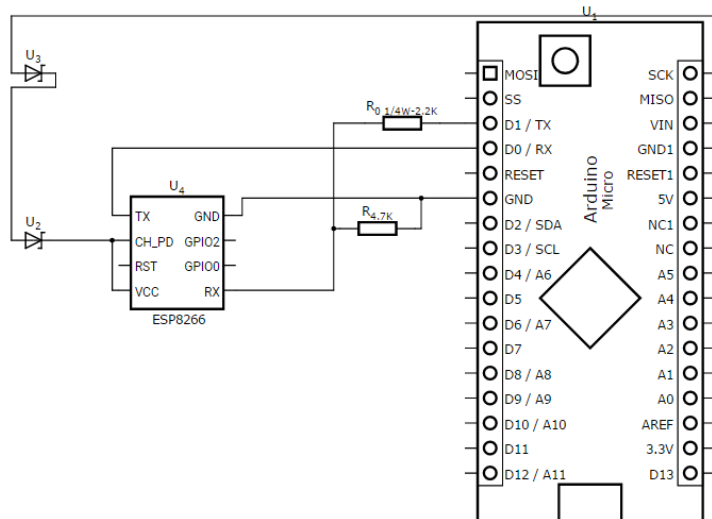


Figura 5. Esquema de connexions ESP- 01

2.1.3.1.3 Prototip

Amb el codi mencionat anteriorment, Arduino estableix una connexió amb dispositiu el ESP8266-01. En canvi, durant les proves del prototip, s'observa que algunes connexions no són satisfactòries, provocant que el mòdul *wireless* es reiniciï per si mateix.

Aparentment, això es degut a un problema del *firmware* que ve instal·lat de fàbrica. Per tant, es requereix l'actualització d'una nova versió la qual pugui reparar aquest problema.

Per tal d'actualitzar-lo cal connectar-lo a una aplicació d'ordinador la qual s'encarrega de transmetre la nova actualització via sèrie. Per realitzar es requereix un convertidor FTDI. Aquest És un convertidor Sèrie-USB (TTL) que permet connectar dispositius TTL per USB.

Així doncs, donat aquest nou element que s'hauria d'adquirir, es realitza una recerca al mercat d'aquests tipus de dispositius i de noves possibles alternatives a la placa mencionada anteriorment.

2.1.3.1.4 Alternatives

Respecte al convertidor sèrie es farà servir, com s'ha mencionat en un primer moment, per l'actualització dels mòduls *wireless*. Aquest dispositius tenen diferent alternatives del xip utilitzat per convertir les senyals. En canvi, l'alternativa que es busca aquí es referent al ESP8266-01.

Per una banda, respecte aquests mòduls hi ha diferent alternatives amb el mateix xip. Per tant, es podria fer ús d'aquests, però encareixen el seu preu, i tenen funcionalitats que no són requerides per aquest projecte.

Per una altra banda, es podria fer ús del mòdul Ethernet ENC28J60. Aquest és un mòdul de baix cost que permet connectar a internet qualsevol microcontrolador amb interfície SPI.

Aquesta última alternativa queda descartada pel fet que hauria de realitzar una instal·lació física de cablejat, per connectar el dispositiu a la xarxa.

Per altra banda, es poden realitzar connexions mitjançant mòduls Bluetooth. Aquest és una gran alternativa, donat que les connexions Bluetooth consumeixen menys que les connexions *WiFi*. En canvi la integració ens era més fàcil a partir del mòdul *WiFi*. Es per aquest motiu, que s'ha optat pel dispositiu mencionat anteriorment.

Finalment, es troben una sèrie de mòduls de desenvolupament anomenats NodeMCU els quals són compatibles amb Arduino. Aquests incorporen les funcionalitats dels microcontroladors emprats per Arduino, amb la incorporació de sèrie del ESP8266. Amb aquesta integració, les llibreries i exemples proveïts pel fabricant faciliten la connexió tant amb el mòdul com amb la xarxa.

Donat que ja ve tot integrat no caldria fer un dispositiu extern per aplicar qualsevol tipus d'actualització dels xips que incorpora aquesta placa. Amb aquesta integració, les llibreries i

exemples proveïts pel fabricant faciliten la connexió tant amb el mòdul con amb la xarxa.

Els preus d'aquestes plaques es aproximat al dels convertidors TTL. Per tant, es considera que tenir una placa amb tot el *hardware* integrat evitarà problemes de compatibilitat, reduiria el número de cables que interconnecten ambdós dispositius i redueix la mida de la placa.

En els punts següents es detallen més característiques d'aquest dispositiu.

2.1.4. NodeMCU v3

NodeMcu són una iniciativa Open Source pel desenvolupament d'un model senzill alhora d'integrar la IOT. Per aquest motiu desenvolupen models de *hardware i software* que faciliti el desenvolupament de programes i aplicacions basats en connexions *Wireless*.

El mòdul té integrat el xip ESP8266ex, una versió revisada del xip mencionat anteriorment, i a més inclou un connector mini USB per programar el microcontrolador i/o comunicar el dispositiu amb el ordinador. Evitant així l'adaptador FTDI a USB.

A més, suporta el mode de desenvolupament d'Arduino mitjançant el *firmware* de la placa i un *plugin* o conjunt de llibreries del IDE d'Arduino anomenat ESP8266.

Les seves característiques^[9] principals són:

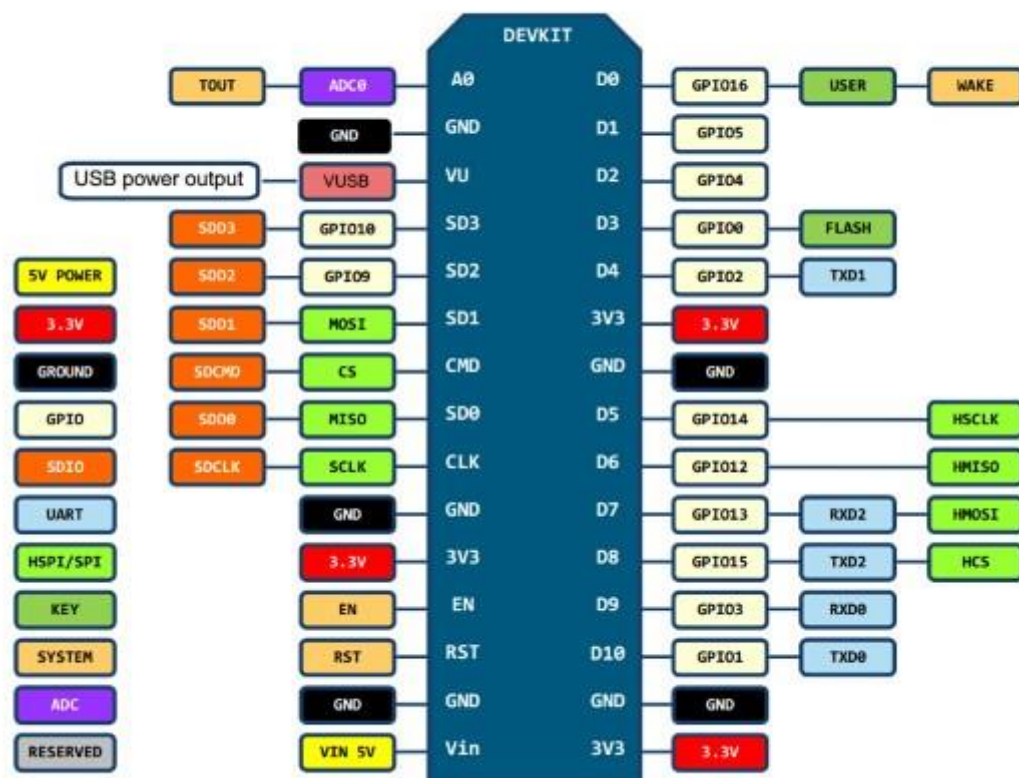
Microcontrolador	ESP8266
Voltatge de funcionament	5V
Voltatge d'entrada recomanat	4-9V
Pins digitals I/O	10
Pins PWM Digital I/O	9
Pins analògics	1
Altres	IIC, comunicació SPI, ADC, altres.
Memòria flash	1MB + 32MB (integrats pel fabricant)
Freqüència del rellotge	80 MHz o 160 MHz
Xip USB-TTL serial converter	CH340G

Llargada	49 mm
Amplada	25 mm
Pes	7.3 grans.

Taula 4. Característiques del NodeMCU v3

A continuació, es mostra el mapa o diagrama d'on està situats cadascun dels pins que incorpora la placa i les seves funcions.

PIN DEFINITION



D0(GPI016) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/ow supported.

Figura 6. Diagrama de pin de la placa NodeMCU v3

2.2. Vàlvula d'entrada

La vàlvula d'entrada serà l'encarregada d'admetre o tallar el pas d'aigua cap al depòsit d'aigua. Aquestes accions seran preses depenent de si el depòsit es ple o no.

2.2.1. Esquema de connexions

La vàlvula treballa a 12V amb una font independent d'Arduino, donat que el voltatge d'Arduino no arriba a aquest voltatge. S'ha optat per vàlvules de 12V per evitar alguns problemes detectats en vàlvules d'inferior voltatges.

Per tal de no fer mal a Arduino, es pot optar per instal·lar un relé o transistor amb resistència. Aquests s'encarregaran de reduir el voltatge que va cap al pin d'Arduino. A més, fent ús d'un díode s'aconsegueix que la corrent circuli en un unit sentit per la vàlvula d'entrada.

L'esquema següent mostra els elements utilitzats i circuit dissenyat per aconseguir el funcionament de la vàlvula.

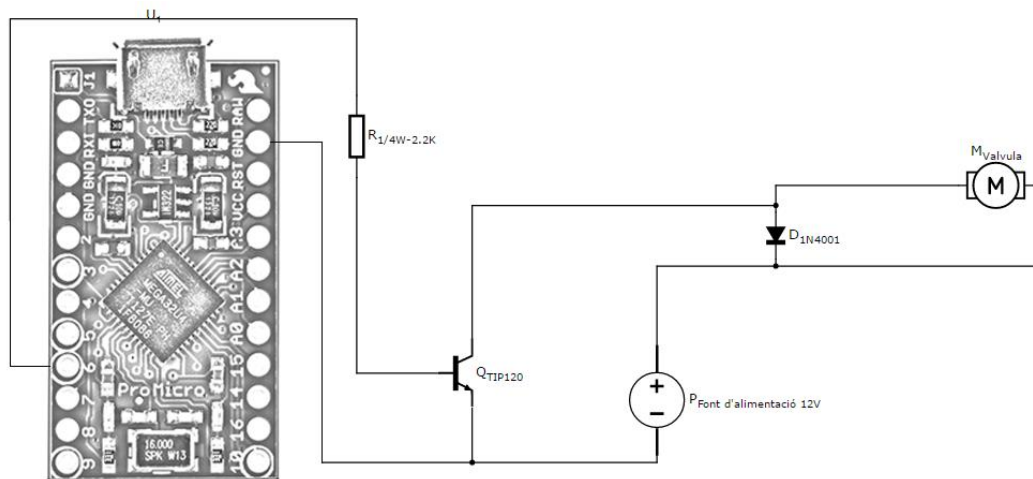


Figura 7. Esquema de connexions de la vàlvula d'entrada

2.2.3 Prototip

Amb el dispositiu totalment operatiu de manera local, es procedeix a la posta en funcionament en un ambient operacional. Aquest no es satisfactori, donat que les vàlvules magnètiques necessiten de certa pressió d'aigua per accionar el seu pas.

Encara que les especificacions compartides pel fabricant indiquen que la pressió mínima és 0 Bars de pressió, no operen de la manera desitjada.

Algunes de les proves realitzades indiquen que, com a mínim, es requereix una caiguda de 60cm per provocar el pas d'aigua per la vàlvula. L'altura a la que es solen col·locar les desguassos es de 45 a 50 cm.

Donat aquesta circumstancia, on les proves realitzades no deixen passar el volum d'aigua requerida, s'opta per replantejar en funcionament del prototip. Així doncs, es prescindeix d'aquest element en la instal·lació.

Això provoca que el tanc d'aigua, de manera obligada, haurà d'estar tancat hermèticament, per tal de que l'aigua que entra no es desbordi.



Figura 8. Vàlvula magnètica

2.2.4 Alternatives

Donat les mencionades circumstàncies en el punt anterior, se estudien altres alternatives de vàlvules. Se investiga la possibilitat de utilitzar unes vàlvules que porten una bola perforada que gira dintre del mecanisme.

Aquest model permet el pas d'aigua sense les limitacions mencionades. En canvi, aquesta queda descartat, donat que el seu cost triplica el de les vàlvules magnètiques. I s'opta, finalment, per selecció d'un altre tipus de tanc el qual veurem més avall.

2.3. Bomba d'aigua

La bomba d'aigua serà l'encarregada de projectar l'aigua que està emmagatzemada dintre del depòsit cap a la cisterna del inodor.

2.3.1 Esquema de connexions

Així com la vàlvula d'entrada, la bomba treballa a 12V amb una font independent d'Arduino. S'ha optat per bombes de 12V per tal d'evitar que les

bombes de 5V es quedin curtes de potencia. A més, com que es farà ús d'una entrada de 12V per la vàlvula d'entrada, es pot fer ús d'aquesta sense haver d'afegir una nova font.

Al igual que la vàlvula d'entrada, es fa ús dels mateixos elements i disseny de circuit.

L'esquema següent mostra els elements utilitzats, connexions i circuit dissenyat per aconseguir el funcionament de la vàlvula.

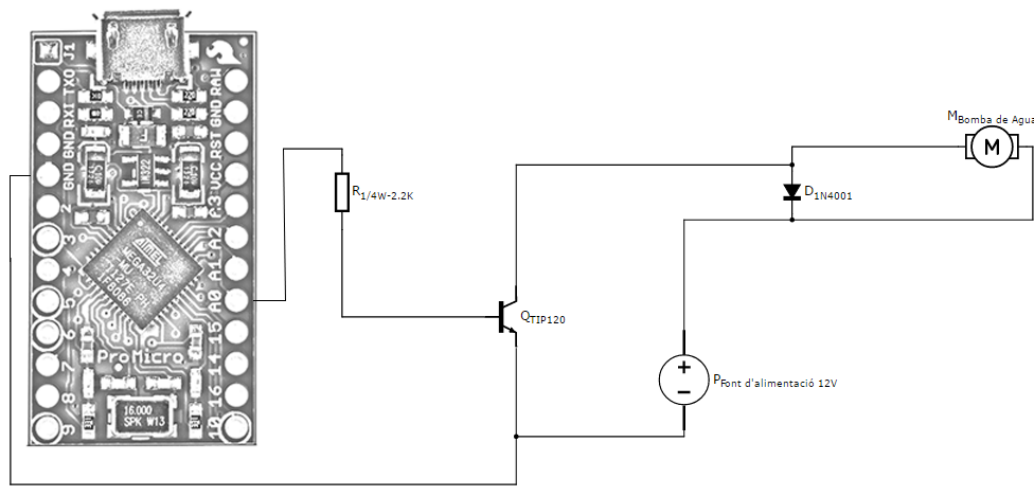


Figura 9. Esquema de connexions de la bomba d'aigua

Fent ús de la funció *WaterPumpON()* se li indica a la placa Arduino que activi la sortida on està connectat la bomba. Provocant així que la bomba s'actioni e impulsi l'aigua cap al inodor.

De manera contraria, amb la funció *WaterPumpOFF()*, Arduino desactiva el pas de corrent provocant que la bomba d'aigua s'aturi.

2.3.3 Prototip

La bomba d'aigua funciona de manera local i en conjunt, una vegada assemblet el prototip. Aquesta no presenta cap problema de funcionament i queda llesta, per ser implantada en el dispositiu final.

Aquesta haurà de posicionar a la part inferior del tanc d'aigua, per tal de que



Figura 10. Bomba d'aigua

l'aigua pugui entrar dintre de la cavitat de la bomba i ,així, pugui ser impulsada per la canonada cap a la cisterna.

2.3.4 Alternatives

La bomba d'aigua és un dels elements primordials dels sistema, i encara que es poden revisar altres alternatives, com bombes amb major potencia, aquest element es essencial per donar simplicitat al projecte.

Com alternativa, s'han revisat la possibilitat de col·locar el depòsit d'aigua a una altura superior a la cisterna d'aigua. D'aquesta manera, la gravetat provoca que l'aigua circuli i entri dintre d'aquesta.

Aquesta solució, inclouria la utilització, a la sortida, d'una vàlvula amb bola perforada, comentada anteriorment i un sensor de flotació. El propòsit d'aquests serà la d'aturar l'ompliment de la cisterna quan aquesta arribi al nivell màxim.

La mencionada alternativa queda descartada, primerament, perquè la comentada vàlvula té un cost elevat, gairebé duplicant el cost de la bomba. Per altra banda, la instal·lació del tanc, estèticament, a certa altura no queda bé, i a més, provoca el redissenyar la tapa superior de la cisterna.

2.4. Vàlvula de tres vies

Les vàlvules de control de tres vies són idònies per a aquelles aplicacions en les quals sigui imprescindible la recirculació de fluid, bé sigui barrejant o diversificant o desviant aquest.

Aquestes vàlvules solen tenir entrada o sortida per AB i sortida o entrada per A y/o per B. Això dependrà de la posició en la que es trobi el motor que fa gira la bola interna.

El propòsit d'aquesta electrovàlvula dintre del projecte, serà la de permetre el pas d'aigua des del tanc o des del pas d'aigua general del domicili. Això dependrà del estat en el que es trobi el tanc.

2.4.1 Esquema de connexions

Aquesta electrovàlvula de tres vies també treballa amb 12V. En canvi, el funcionament i les connexions varien en respecte a la vàlvula magnètica.

En aquest cas, l'electrovàlvula té tres connexions. Per una part es troba la connexió del *ground* i per l'altra trobem dos connexions que indiquen en quina posició s'estacionarà la vola que desvia el caudal d'aigua.

L'esquema següent mostra els elements utilitzats, connexions i circuit dissenyat per aconseguir el funcionament de la vàlvula.

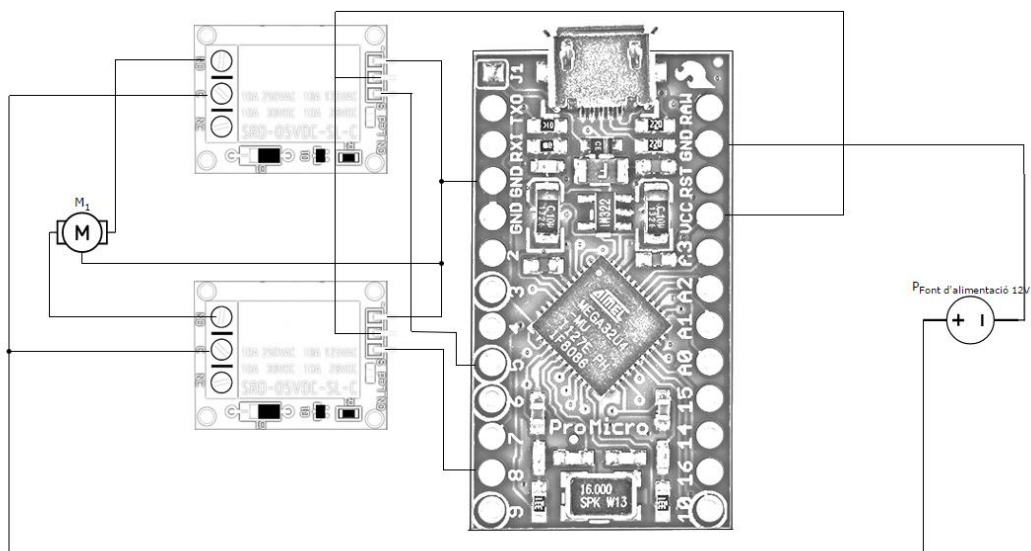


Figura 11. Esquema de connexions de la vàlvula de tres vies

Durant les proves del prototip i assemblatge del prototip ha quedat desestimat aquesta alternativa. Posteriorment, s'especifica el motiu pel qual ha quedat descartat.

Finalment s'ha optat per la incorporació de dos relés. A continuació, queden referenciats els elements utilitzats, connexions i circuit dissenyat per aconseguir el funcionament de la vàlvula.

2.4.3 Prototip

Aquesta vàlvula ha estat afegida una vegada el prototip final estava en funcionament. Inicialment, el prototip no contemplava aquesta opció donat que es volia treballar de manera independent del subministrament principal.

En canvi, malgrat que el tanc es suficient gran com per emmagatzemar suficient aigua, això podria presentar un risc i deixar sense subministrament a la cisterna del vàter.

Així doncs, en el remot cas que el tanc quedi sense aigua, aquesta vàlvula obrirà el pas principal i el WC treballarà de manera estàndard.

2.4.4 Alternatives

De la mateixa manera que s'ha comentat anteriorment, amb la vàlvula d'entrada es podien haver utilitzat electrònicament vàlvules magnètiques o de boles (com ha estat aquest cas) de tres vies.

Finalment, les vàlvules magnètiques han quedat descartades per evitar problemes, com el mencionat al apartat de la vàlvula d'entrada. Encara que, en aquest cas, probablement si hagués funcionat perquè la bomba d'aigua genera suficient pressió com per provocar el pas d'aigua.

2.5. Sensor de nivell

El Sensor de nivell és un dispositiu electrònic que mesura l'altura del líquid, dins de la cisterna. Aquest marca l'altura de l'aigua sobre un nivell preestablert.

En aquests sensors de nivell de punt, un flotador magnètic es mou en la superfície del líquid, accionant un segellat hermèticament *reed switch* en la tija del dispositiu.

Aquest funciona amb un interruptor que activa a Arduino i l'indica quan el nivell d'aigua, dintre de la cisterna, ha pujat o baixat. En aquest últim cas, Arduino activarà la bomba fins que el sensor de nivell torni a pujar al seu estat inicial.

2.5.1 Esquema de connexions

El funcionament del sensor de nivell es igual o similar al que té un pulsador. Aquest té dues connexions el qual envien o no les senyals depenent de la posició del flotador.

L'esquema següent mostra els elements utilitzats, connexions i circuit dissenyat per aconseguir el funcionament de la vàlvula.

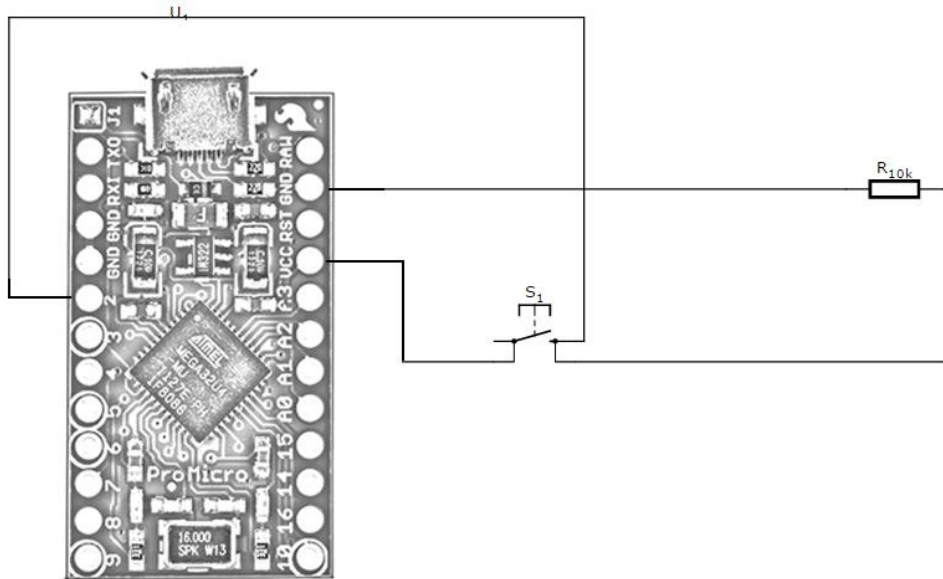


Figura 12. Esquema de connexions del sensor de nivell

2.5.3 Prototip

Aquesta vàlvula ha estat afegida una vegada el prototip final estava treballant. Inicialment, el prototip no contemplava aquesta opció donat que es volia treballar de manera independent del subministrament principal.

En canvi, malgrat que el tanc es suficient gran com per emmagatzemar suficient aigua, això podria presentar un risc i deixar sense subministrament a la cisterna del vàter.

Així doncs, en el remot cas que el tanc quedi sense aigua, aquesta vàlvula obrirà el pas principal i el WC treballarà de manera estàndard.

2.5.4 Alternatives

Donat que, en aquest cas, es vol saber quan el nivell d'aigua, dintre de la cisterna, arriba al nivell màxim es poden trobar diferent alternatives al sensor de flotació.

Per una banda, els sensors capacitius tenen la facultat de detectar elements sense requerir l'entrada en contacte amb aquests. Aquests detecten alteracions en el camp magnètic provocats per certs elements. D'aquesta manera es pot saber quan l'aigua baixa de nivell i puja de nivell dintre de la cisterna.

El grossor de la ceràmica de la cisterna pot arribar a donar falses deteccions o no detectar correctament. Es per aquest motiu que no es contempla aquesta alternativa.

Per altra banda, el sensor d'humitat el qual consisteix en dues plaques separades entre si per una distància determinada.

Aquestes plaques estan recobertes d'una capa de material conductor. En el cas que l'aigua arribi fins aquest material conductor es crearà un pont entre una punta i una altra, la qual cosa serà detectat per un circuit de control amb un amplificador operacional. Aquest amplificador serà l'encarregat de transformar la conductivitat registrada en un valor analògic que podrà ser llegit mitjançant Arduino per saber si s'està o no produint una baixada del volum d'aigua dintre de la cisterna.

2.6. Tanc d'aigua

El tanc d'aigua es l'encarregat d'emmagatzemar l'aigua rebutjada del rentamans. Aquest és un dipòsit hermètic que permet que l'aigua, en arribar al seu nivell màxim, no es desbordi i l'aigua restant es desviï cap al clavegueram.

Donat que quan s'acciona la cisterna es rebutja entre 3 i 10 litres d'aigua, depenent de l'eficient que sigui la cisterna i la seva vàlvula de descarrega, es recomana que el tanc tingui una capacitat mínima de 10 litres. En cas contrari, per cada descarrega el tanc quedarà buit per complet i necessitarà del pas d'aigua general per tal d'omplir la cisterna.

La mida d'aquest element queda condicionat al espai i posicionament que tindrà l'element dintre del servei o lloc o on es vol realitzar la instal·lació. En aquest cas, s'ha seleccionat un dipòsit de 25 litres, el qual té una mida de 25x34x40 cm, suficientment gran per emmagatzemar aigua necessària per descarregar dintre d'una cisterna estàndard (aproximadament uns 6 litres per descarrega) una mitja d'entre 3 i 5 vegades, sense haver de tornar-la a omplir de nou.



Figura 13. Tanc d'aigua

2.6.1. Disseny

Aquest element té identificat un forat d'entrada i una de sortida, per on entra i surt l'aigua. Aquests tenen un diàmetre de 15mm per tal de poder instal·lar el maniguet d'unió de 1/2".

L'entrada es col·loca a la part superior del dipòsit. Aquesta no ha de superar l'altura a la qual està col·locat el desguàs del rentamans. En el cas de superar l'altura indicada, el dipòsit no s'omplirà per complet. El qual significa malgastar part de l'espai.

Per contra, la sortida, on se instal·la la bomba, s'ha de instal·lar a la part inferior. En cas contrari, l'aigua que queda per sota de la boca d'entrada de la bomba quedarà estancada.

2.6.2. Alternatives

La selecció del tanc ha quedat determinades per les següents circumstàncies, encara que n'hi ha certes alternatives, explicades a continuació, que poden solucionar les problemàtiques trobades.

En primera instància, s'han realitzat proves amb tanc obert en la part superior. Dintre d'aquest camp es troben elements bàsics com cubs d'aigua. Aquesta alternativa requereix del control del volum d'aigua dintre del recipient. En cas contrari, l'aigua desbordaria fora d'aquest.

Per tant, aquesta alternativa requereix d'una vàlvula d'entrada la qual bloqueja el pas d'aigua cap al dipòsit. A més, es requereix saber en quin nivell es troba l'aigua dintre del tanc. Així doncs, s'hauria de incorporar un sensor que indiqui quan el recipient es ple. Com exemple, podem destacar els sensors de flotació mencionats anteriorment.

Com s'ha mencionat en el punt referent a la vàlvula d'entrada, les vàlvules utilitzades requereixen d'una pressió mínima per poder permetre el pas. Per tant, aquesta alternativa ha quedat descartada.

Com a segona alternativa, i basat en la primera proposta, s'ha de indicar que hi ha la possibilitat de treballar amb tancs oberts sense necessitat de tenir una vàlvula d'entrada i un sensor de flotació.

Aquesta alternativa, requereix que l'entrada d'aigua es situï a la mateixa altura que el desguàs, d'aquesta manera quan el tanc arriba al seu nivell màxim, gràcies a la gravetat, no es deixa passar més aigua i es rebutja cap al desguàs principal.

2.7. Sensor de flux d'aigua

El sensor de flux d'aigua és un dispositiu que incorpora un molinet que en fer-li passar l'aigua a través d'ell, es pot saber quanta aigua està sent enviada cap a la cisterna.

El dispositiu fa ús de l'efecte Hall per comptabilitzar l'aigua. Aquest efecte consisteix en l'aparició d'un camp elèctric en un conductor quan és travessat per un camp magnètic.

En aquest cas, el molinet té un petit imant que quan es fa gira genera impulsos que són detectats per un sensor magnètic que incorpora el dispositiu, amb el qual es pot mesurar la quantitat de voltes que el molinet ha realitzat.



Figura 14. Sensor de flux d'aigua

2.7.1 Esquema de connexions

El sensor ve amb tres cables: vermell 5-24VDC, negre (terra) i groc. Aquest últim serà el que enviarà els polsos detectat gràcies al efecte Hall.

Per evitar problemes de detecció i que el voltatge no variï, es col·loca una resistència, la qual evitarà danyar la placa d'Arduino i comptabilitzar valors erròniament.

L'esquema següent mostra els elements utilitzats, connexions i circuit dissenyat per aconseguir el funcionament del sensor.

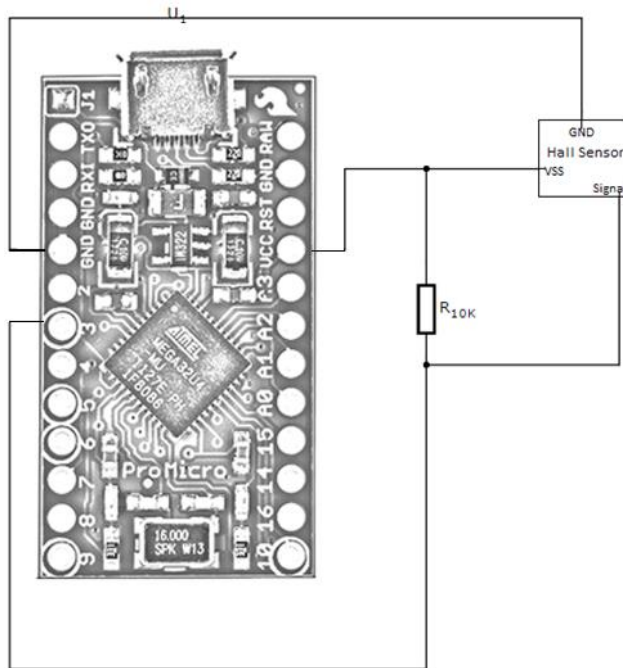


Figura 15. Esquema de connexions del sensor de flux d'aigua

2.7.3 Prototip

Després de les proves realitzades i el calibratge realitzat al dispositiu, el sensor comptabilitza el pas de fluids. Aquest es connecta a la sortida de la bomba, la qual quan entra en funcionament i comença a impulsar l'aigua cap a la cisterna, provocant el moviment del molinet i la posterior detecció del sensor. El qual envia els impulsos cap a la placa Arduino i, mitjançant el codi anterior, es calcula el consum d'aigua.

2.7.4 Alternatives

Aquest dispositiu no presenta grans alternatives, donat que el funcionament es semblant en tot ells. En major mesura, i depenent dels requeriments que es requereixen per cada instal·lació, es podria utilitzar altres models amb major o menor qualitat.

2.8. Assemblatge del Prototip

Una vegada està definit cadascú dels elements que compondrà aquest projecte, es procedeix a un primer assemblatge tant dels dispositius com del codi, i es realitzen proves de funcionament del conjunt.

La primera versió d'aquest prototip contemplava la instal·lació d'un depòsit obert amb una vàlvula d'entrada el qual s'accionava quan arribava al nivell màxim.

Com s'ha mencionat anteriorment, aquest sistema ha quedat descartat perquè la vàlvula d'entrada necessitava certa pressió per admetre el pas

d'aigua cap al depòsit. Així doncs, l'opció escollida pel dispositiu final ha estat un tanc tancat hermèticament i així s'evita tenir que treballar amb aquest tipus de vàlvula. Amb aquesta opció s'elimina, també, l'haver de fer ús d'un sensor de nivell donat que no requereix obrir o tancar la vàlvula.



Figura 16. Prototip inicial



Figura 17. Prototip inicial - Sensor de flux

Aquest sensor de nivell, s'aprofita i se instal·la dintre de la cisterna del inodor per tal de saber quan es produeix una descarrega de l'aigua que aquesta conté. Així, quan aquest esdeveniment succeeix, el sistema activa la bomba i la vàlvula de tres vies deixa passar l'aigua des del tanc cap a la cisterna.

Aquesta circulació d'aigua produeix l'activació de la funció que comptabilitza els polsos, el qual es generat pel moviment de l'aigua. A partir d'aquí, el sistema s'extreu els resultats e indica el número de litres gastat del tanc i traspassats cap a la cisterna.

El prototip ha quedat funcionant operativament i, a partir dels resultats extrets, s'ha pogut veure les necessitats que el dispositiu final hauria de tenir. Com per exemple, com s'ha comentat anteriorment, l'eliminació de la vàlvula d'entrada i el sensor de nivell que anava instal·lat dintre del depòsit d'aigua.

Així, també, en aquesta fase ha quedat constatat que es requeria una millora per tal de poder accedir a les dades sense haver d'accedir al dispositiu. I així poder realitzar el posterior estudi de manera més detallada.

Per tant, abans de començar a assemblejar el producte final, s'ha realitzat una recerca al mercat, tal com s'ha comentat abans. I s'han revisat les mencionades dues alternatives. Per un costat, la implantació del mòdul ESP8266-01 a la placa d'Arduino Micro Pro i per l'altra banda, l'adaptació del codi al element Node MCU v3.

2.9. Producte final

Aquest últim bloc no defereix en gran mesura del punt anterior. En aquesta fase, s'han procedit a la instal·lació de cadascun dels elements mencionats anteriorment, incloent-hi el tanc tancat hermèticament.

A més, es procedeix a la realització de la instal·lació de tots els elements que componen el dispositius i l'electrònica.

Aquest procés comporta, primerament, la connexió del tub per on es recupera l'aigua des del rentamans.



Figura 19. Bomba d'aigua instal·lada

A continuació, a l'altre extrem del tanc d'aigua

es procedeix a la inserció de la bomba d'aigua encarregada d'impulsar l'aigua que hi ha dintre del tanc cap a la cisterna del inodor.

A partir d'aquí, l'aigua que es impulsada des de la bomba arriba al sensor de flux, el qual envia al microcontrolador els impulsos generats per la circulació d'aigua. I, just a continuació, l'aigua traspasa la vàlvula de tres vies, la qual està connectada directament a l'entrada d'aigua de la cisterna del inodor.

Aquestes interconnexions ha requerit l'adquisició i la posterior instal·lació de connectors i elements propis de la fontaneria.

Per últim, deixant de banda els elements que transporten l'aigua, es realitza la instal·lació del sensor de nivell d'aigua dintre del inodor.



Figura 18. Tanc d'aigua instal·lat



Figura 20. Sensor de flux i fontaneria

Tots els dispositius quedaran connectats amb el circuit electrònic dissenyat per aquest propòsit. L'esquema següent mostra com s'ha dispost cadascú d'ells i així com el circuit dissenyat per aconseguir el funcionament.

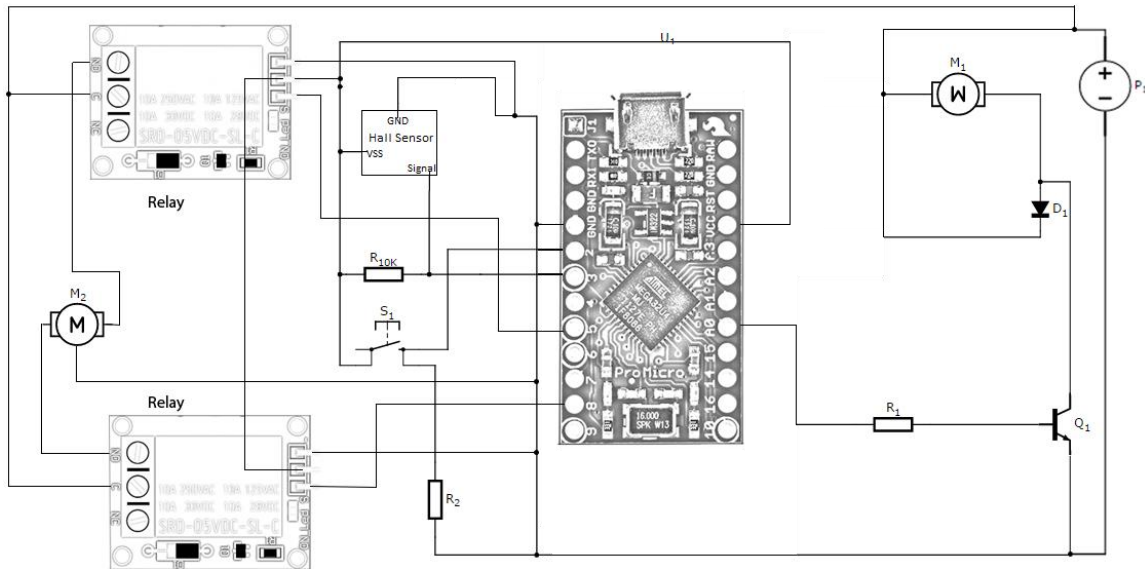


Figura 21. Diagrama de connexions generals

3. Millores

L'apartat de millores incorpora les funcionalitats mencionades anteriorment per la integració d'un mòdul *Wireless* i la possibilitat de poder transmetre les dades recuperades des del mòdul Arduino.

3.1. Integració del mòdul NodeMCU

Per tant, una vegada presentat el mòdul NodeMCU i feta la instal·lació del producte final, es procedeix a veure com quedaria la nova configuració del programa i del circuit electrònic.

Per tant, en una primera instància, es distribueixen cadascuna de les connexions i elements que pertanyen als dispositius mencionats en els punts anteriors. A continuació, es mostra el diagrama o circuit de connexions per posar en funcionament el producte final amb el nou microcontrolador.

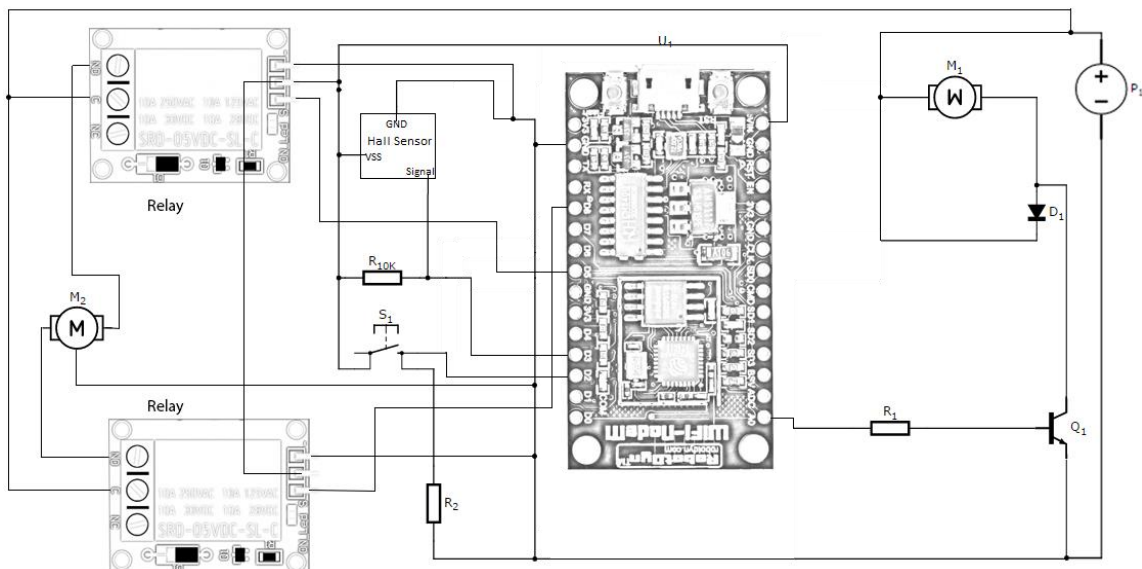


Figura 22. Diagrama de connexions amb el mòdul NodeMCU

En segona estància, des de l'IDE d'Arduino es descarrega les llibreries pertanyent a la nova placa. D'aquesta manera, si es truquen a les llibreries del NodeMCU des del programa que s'està desenvolupant, s'estalvia l'escriptura de noves funcions per tal de crear i establir connexió amb els pins i el mòdul *WIFI* que porta incorporat.

```
3 //Libraries//
4 #include <ESP8266WiFi.h>
5 //////////////////////////////////////////////////
```

Figura 23. Definició llibreria ESP8266Wifi.h

A continuació, es crearan les variables que indiquen el SSID i la contrasenya de la connexió *WIFI* a la que es connectarà la placa.

```
6 //configuracion wifi //////////////////////////////////////
7
8 const char* ssid      = "SSID Conexio Wifi"; //SSID Conexio Wifi
9 const char* password = "Pass Conexio Wifi"; //Pass Conexio Wifi
10
11 //////////////////////////////////////
```

Figura 24. Definició dels detalls de la connexió WIFI

Amb aquestes dades definides, es crearà la funció de connexió *wireless* a la xarxa. Definim això com una funció, es podrà realitzar la connexió amb el *router* al inici, és a dir, en el `Setup()`, o cada vegada que es requereixen enviar dades a la xarxa. Amb aquesta última, al tenir l'antena apagada durant la major part del temps, el consum del dispositiu es reduirà.

```
13 void conexion_wifi()
14 {
15
16     Serial.println();
17     Serial.println();
18     Serial.print("Connectant a ");
19     Serial.println(ssid);
20
21     WiFi.begin(ssid, password);
22
23     while (WiFi.status() != WL_CONNECTED) {
24         delay(500);
25         Serial.print(".");
26     }
27
28     Serial.println("");
29     Serial.println("WiFi connectat");
30     Serial.println("Direcció IP: ");
31     Serial.println(WiFi.localIP());
32 }
```

Figura 25. Codi de connexió WIFI

3.2. Enviament de dades

A continuació, es procedeix a definir com i a on s'envien les dades generades pel sensor de flux. Per tal de processar les dades, a part d'Arduino per enviar les dades, s'ha fet d'un entorn de desenvolupament web en el qual es poden crear aplicacions mitjançant el servidor web Apache, el gestor de base de dades de mysql i el llenguatge de programació PHP.

3.2.1 Base de dades

El primer que s'ha realitzat ha estat la definició de la base de dades i les taules que contindran els registres. En aquest cas, la taula on s'inseriran els registres del comptador serà la taula '*registro*'. Aquesta conté dues variables, la

primera es la data en qual es va inserir el registre en base de dades i la segona contindrà el valor capturat des del sensor de flux.

La següent imatge, mostra, com s'ha indicat al gestor de base de dades MySQL, la creació de la taula i les seves columnes.

```
25  --
26  -- Estructura de tabla para la tabla `registro`
27  --
28
29  CREATE TABLE `registro` (
30    `fecha` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
31    `valor` decimal(10,2) NOT NULL
32  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
33
```

Figura 26. Creació de la taula de registres

Amb aquesta taula ja es sufficient com per rebre la informació i poder registrar-la, pel posterior anàlisis.

Veure Annex IV per veure com s'ha gestat la creació de la base de dades.

3.2.2 PHP

PHP és un llenguatge de codi obert molt popular, adequat per a desenvolupament web i que pot ser incrustat en HTML.

Aquest llenguatge es processa en servidors. Això vol dir que des del client s'envien les dades amb una sol·licitud al servidor. Aquest les processa, reuneix les dades i les retorna com si fossi una pàgina HTML estàtica.

En aquest cas, s'ha dissenyat el codi de la següent manera:

```
1  <?php
2      // Connexio amb MySQL
3      include("config.php");
4
5      // Sentencia INSEER SQL amb les dades enviades des de Arduino
6      $sql = "INSERT INTO registro (valor) VALUES ('".$_GET["valor"]."')";
7
8      // Execucio de la sentencia SQL
9      mysqli_query($db,$sql);
10     mysqli_close($db);
11
12     // Per tal de revisar que les dades s'han insertat fer us de review_data.php (op
13     //header("Location: review_data.php");
14  ?>
```

Figura 27. Codi de connexió amb MySQL

Com es veu en la imatge anterior, primerament es defineix els paràmetres de la base de dades i es procedeix a connectar amb aquesta.

A continuació, es defineix la sentència “INSERT”. Aquesta agafarà el valor que Arduino indiqui i procedirà a inserir la informació dintre de la tabla de registres.

Per últim, s’executa la sentència i es tanca la connexió amb la base de dades.

3.2.3 Execució des d’Arduino

Una vegada es té definit el lloc on es guardaran les dades i el mètode a utilitzar per inserir aquestes dades, es procedeix a indicar-li a Arduino com executar aquesta funcions.

En primera estància, la funció definida intenta contactar amb el servidor web per veure si n’hi ha connexió i si es pot enviar el paquet de informació.

```
2 void conexion_portal()
3 {
4     delay(5000);
5
6
7     Serial.print("Connexió amb: ");
8     Serial.println(host);
9
10    WiFiClient client;
11    const int httpPort = 8080;
12    if (!client.connect(host, httpPort)) {
13        Serial.println("connection failed");
14        return;
15    }
16
```

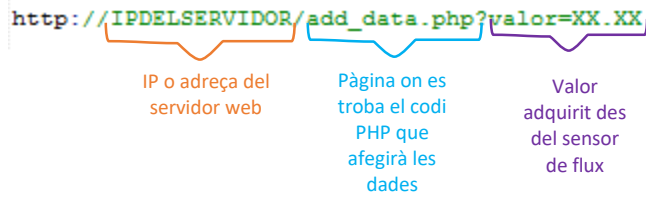
Figura 28. Arduino connectant amb el portal

En segona estància, si n’hi ha connexió i amb les dades obtingudes des del sensor flux, es procedeix a registrar la informació. Per tal de realitzar-ho es fa ús del mètode GET.

Aquest és un mètode utilitzat en HTML el qual, indicant-li les variables a enviar, transmet informació a través de la direcció URL.

Per tant, si ja es té definit el codi PHP que espera l’enviament d’una variable i Arduino té el resultat del comptador d’aigua, es s’haurà de indicar a Arduino on està el servidor web i quina és la pàgina a executar.

En certa mesura, Arduino executarà la sentència web:



La imatge que ve a continuació conté el codi explicat anteriorment i com s'ha aconseguit la construcció i enviament de les variables.

```

17 //Creacio de la URL
18 String url = "/add_data.php?";
19 url += "valor=";
20 url += litros;
21
22
23
24 Serial.print("Sollicitant URL: ");
25 Serial.println(url);
26
27 //Trucara al codi php que afegeix data en DB i enviara el valor de litres estalviat
28 client.print(String("GET ") + url + " HTTP/1.1\r\n" +
29 "Host: " + host + "\r\n" +
30 "Connection: close\r\n\r\n");
31 unsigned long timeout = millis();
32 while (client.available() == 0) {
33     if (millis() - timeout > 5000) {
34         Serial.println(">>> Client Timeout !");
35         client.stop();
36         return;
37     }
38 }
39

```

Figura 29. Codi de creació de la URL

3.3. La interfície d'usuari

Per tal de presentar les dades visualment més atractives pel usuari, s'ha dissenyat una aplicació web la qual llegeix la informació registrada en base de dades i la presenta en forma de gràfiques en la web.

Per tal de desenvolupar aquesta eina s'ha fet ús de la llibreria Chart.js, el mencionat llenguatge de programació de PHP, AJAX i JSON.

3.3.1. AJAX

AJAX és una tècnica que permet la comunicació asíncrona entre un servidor i un navegador en format XML mitjançant programes escrits en Javascript.

El principal objectiu de l'AJAX, és intercanviar informació entre el servidor i el client (navegadors) sense la necessitat de recarregar la pàgina.

D'aquesta forma, es guanya en usabilitat, experiència i productivitat del usuari final.

3.3.2. JSON

JSON és un format lleuger per als intercanvis de dades, bàsicament JSON descriu les dades amb una sintaxi dedicada que s'utilitza per identificar i gestionar les dades.

3.3.3. Chart.js

Chart.js és una llibreria *javascript* que permet generar diferents tipus de gràfiques. Aquesta llibreria es *open source* baix llicència MIT i disposa de molt bona documentació amb exemples d'ús molt complets.

3.3.4. L'aplicació web

Una vegada ja es té creat cadascun dels gràfics i definida la informació què es vol presentar en pantalla, es procedeix a realitzar la disposició de totes elles en la nostra aplicació.

La primera part que es troba en el web el requadre que conté el sumari d'aigua estalviada. Aquest sumari indica quin es el número de litres estalviats en total, així com el número de litres estalviats avui, el mes i l'any en què es troba .



Total estalviat	Avui	Aquest mes	Aquest any
196.93	1.20	193.21	196.93

Figura 30. Sumari de litres estalviats

Aquesta informació ha estat obtinguda de la mateixa manera que s'han obtingut els gràfics, és a dir, fent ús del script mencionat anteriorment.

La segona part, incorpora les gràfiques que s'ha dissenyat mitjançant l'script i la llibreria Chart.js.

La gràfica següent mostra mes a mes, durant l'any en curs i el volum d'aigua estalviada. Aquesta informació, es pot veure il·lustrat en forma de barres en la gràfica.

A més, en aquest cas, és una gràfica mixta on s'ha afegit una línia de punts per tal de indicar el número de descarregues o, dit d'una altra manera, el número de vegades que es pressionarà el botó de la cisterna cada mes.

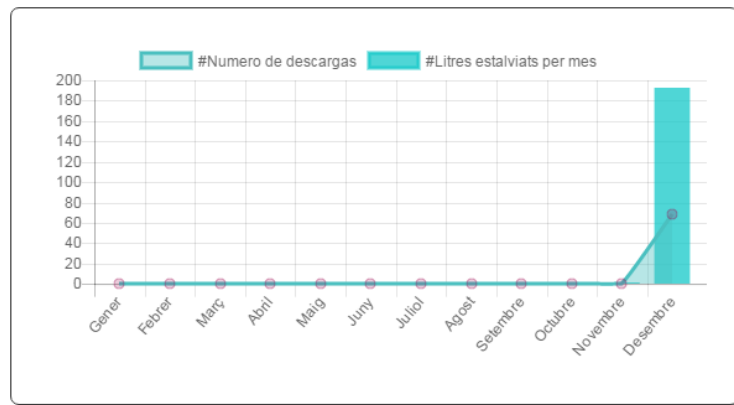


Figura 31. Gràfica de litres recuperats mensualment

A continuació, es mostra una gràfica de línia de punts la qual indica, durant el mes en curs i el número de litres d'aigua estalviats diàriament.

Cal destacar que, en ambdues gràfiques, quan el cursor es posiciona a sobre de qualsevol punt o barra, presenten la informació del número de litres estalviats referent a aquell dia o mes.

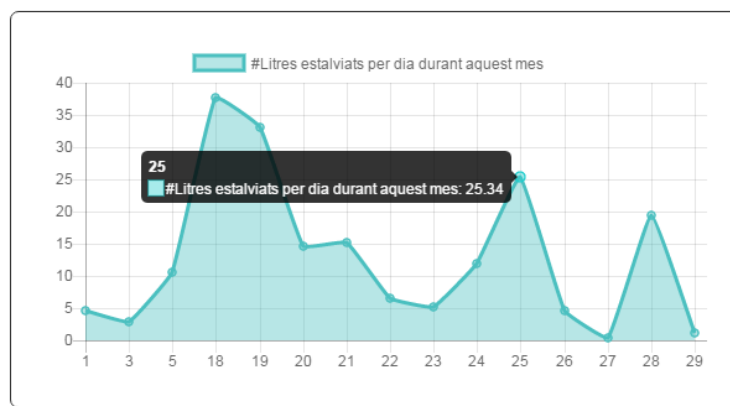


Figura 32. Gràfica de litres recuperats diàriament

La disposició e interfície final de l'aplicació web es presenten en la següent imatge:

Total estalviat	Avui	Aquest mes	Aquest any
196.93	1.20	193.21	196.93

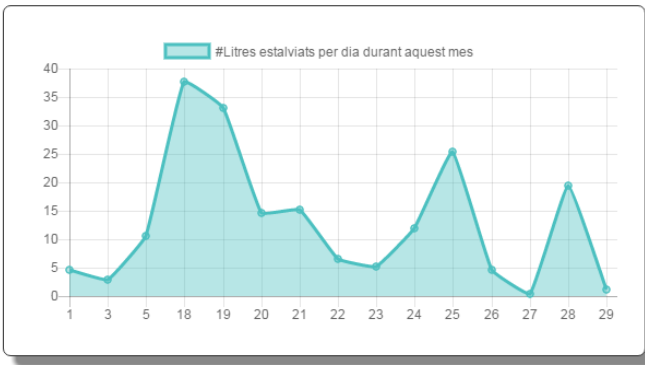
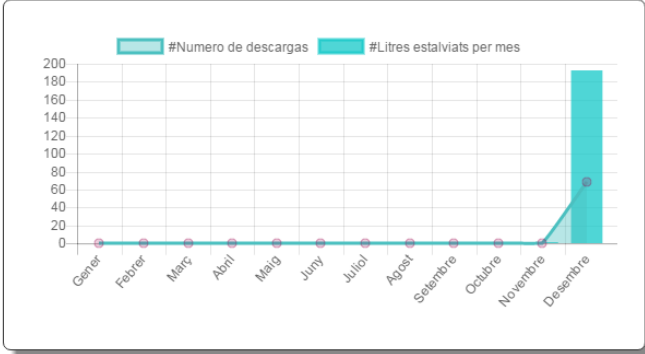
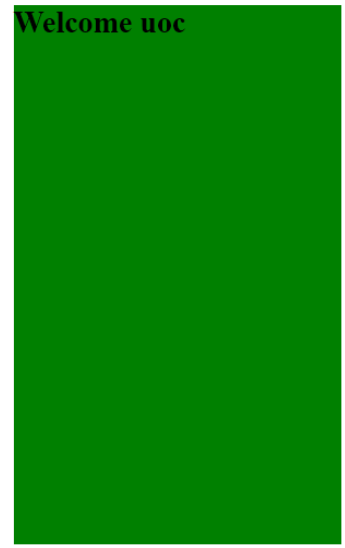


Figura 33. Aplicació Web

Per últim, amb la idea de que es puguin connectar més d'un dispositiu a l'aplicació web i, per tant, més d'un usuari, s'han definit noves taules en la nostra base de dades per tal d'administrar-hi l'accés al portal.

Aquest accés ve gestionat per una pàgina de inici on els usuaris s'hauran de indicar el seu usuari i la contrasenya. Si es correcte, el codi donarà accés a consultar les dades sobre l'estalvi d'aigua al usuari en qüestió.

The image shows a login form titled "Registre" (Register) in a dark grey header. Below the header, there are two input fields. The first is labeled "Nom d'usuari" (Username) and contains the text "uoc". The second is labeled "Contrasenya" (Password) and contains three dots "•••". Below these fields is a blue button labeled "Connectar" (Connect).

Figura 34. Accés a l'aplicació Web

4. Resultats

Després de les diferents proves i validacions, es procedeix a la implantació del dispositiu en un emplaçament habilitat amb el que poder realitzar les diferents proves en situacions reals.

A més, mitjançant la interfície gràfica presentada en el punt anterior, es pot fer un anàlisi dels resultats obtinguts de la utilització del sistema. Per tant, a continuació es detallen els resultats obtinguts de la utilització del sistema durant un període de, aproximadament, 2 setmanes.

El dispositiu ha estat en complet funcionament durant el dia 18 de Desembre i el 29 del mateix mes. En un plantejament on dues persones han realitzat una utilització en actiu del dispositiu.

Una vegada transcorregut aquest període mitjançant el gràfic següent disponible en la pàgina web, i el qual ha estat explicat en el punt anterior, es pot visualitzar diàriament el número de litres estalviat.

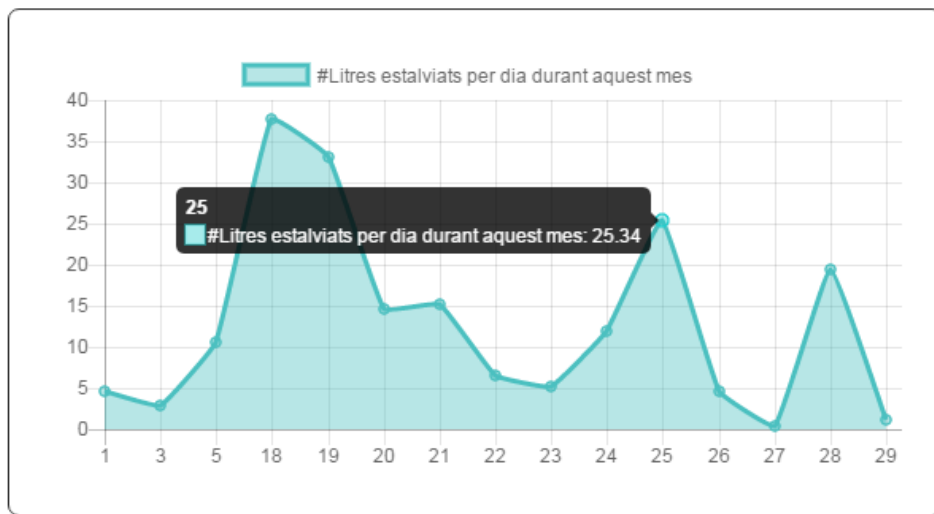


Figura 35. Gràfica de litres estalviats diàriament

A continuació es detallen aquest resultat, en la taula següent, on es mostra amb més detall el número de litres estalviats a diari.

Data	Litres
18/12/2016	37.72
19/12/2016	33.08
20/12/2016	14.6
21/12/2016	15.11
22/12/2016	6.56
23/12/2016	5.23
24/12/2016	11.95
25/12/2016	25.34
26/12/2016	4.52
27/12/2016	0.32
28/12/2016	19.35
29/12/2016	1.2
Total	174.98

Taula 5. Resultats número de litres estalviats diàriament

Segons aquest resultat es pot observar que el sistema ha estat capaç de recuperar i estalviar un total de 174.98 litres. Indicant, per tant, que diàriament s'ha estalviat una mitjana de 15.91 litres.

A partir d'aquí, s'ha de tenir en compte que un any té un total de 365 dies, aquesta instal·lació podria estalviar, anualment, 5.807 litres.

Amb aquests resultats i el cost relacionat a la compra de cadascun dels dispositius necessaris per a la realització del dispositiu, el qual és de 108,75 € i sense tenir en compte el temps de instal·lació, el número d'anys mínim en el que aquesta instal·lació pot recuperar la inversió és, aproximadament, 9 anys.

Aquesta dada s'ha obtingut comparant les dades indicades en la hipòtesi inicial al punt 1.3., on s'indicava el següent:

“cada família estàndard de 4 membres està rebutjant cada any, de mínim, una mitjana de 21.900 litres i un màxim 116.800 litres. Així es pot conèixer que per família, tenint en compte que el preu de l'aigua es pròxim al 2 euros per metre cúbic, el inodor suposa un cost de 42€ a 233€ euros cada any”

En el cas dels 4 membres s'hauria de realitzar un anàlisi més detallat per veure el número de litres que una família estàndard podria estalviar. Aquesta variable s'hauria

d'obtenir tenint en compte el número de litres de capacitat que té el depòsit. I, a més s'hauria de tenir en compte que a més persones més aigua podria anar cap al depòsit. Per tant, el número de litres estalviats anualment seria superior i la inversió seria recuperada més ràpid que la prova realitzada.

5. Conclusions

Donats els resultats anteriors, s'anota que s'han de tenir en compte les diferents variables, com poden ser el volum d'aigua que té el dipòsit, l'eficiència de la cisterna del inodor, el número de persones; per tal de determinar la viabilitat del projecte.

Amb aquestes variables es pot concloure que perquè sigui viable i, a la vegada, es pugui estalviar més aigua, es molt necessari instal·lar una cisterna eficient que no rebutgi més de 4 litres.

El volum d'aigua del tanc es recomanable utilitzar el mateix indicat en les proves, però això, com s'ha mencionat en apartats anteriors, pot variar depenent del espai que es té a la instal·lació.

A més, externament aquest projecte, és aconsellable la instal·lació d'una segona entrada d'aigua provinent, per exemple, de la dutxa. D'aquesta manera el número de litres d'aigua estalviats seria major.

Aquest últim comentari ve donat de l'observació realitzada dels resultats, on s'observa que hi ha dies on l'aigua recuperada del rentamans no arriba suficient per omplir la cisterna utilitzada i s'ha de fer ús de l'entrada d'aigua principal.

Revisió del objectius

A partir d'aquest punt i l'anàlisi inicial realitzat en els paràgrafs anteriors, es pot observar que el primer objectiu principal per recuperar les aigües del rentamans s'ha assolit donat que el dispositiu té l'habilitat d'emmagatzemar l'aigua i enviar-la a la cisterna del inodor per la posterior descàrrega.

En canvi, el segon objectiu principal es pot observar que n'hi ha moltes variables a tenir en compte, les quals indiquen que, en certes instal·lacions, seria viable i altres seria difícil recuperar la inversió inicial ja que requerirà la instal·lació d'elements secundaris.

En referència als objectius secundaris mencionats, es pot analitzar que s'han ideat i desenvolupat les especificacions tècniques del producte per tal de posar en funcionament una instal·lació. En canvi, si que es cert que aquest punt podria haver incorporat més documentació e informació del codi com de la pròpia electrònica. Aquest comentari ve donat perquè es donen coses per sobre enteses i pot haver usuaris que no sàpiguen del que s'està comentant.

Pel que fa referència a cadascú dels dispositius que componen la instal·lació s'han elaborat les proves pertinents i s'han verificat les diferents funcions que cadascú d'ells poden aportar tant interna com externa a aquest dispositiu.

En relació a les funcions internes, una vegada realitzada les operacions individual dels dispositius, la implantació del producte es més assequible ja que aquesta es modular. Així doncs, únicament, cal interconnectar el codi dels dispositius i en el cas que un tercer usuari vulgui implantar el producte o modificar-lo es més senzill d'entendre.

Finalment, sobre les revisions de les conclusions, es pot dictaminar que el producte, una vegada instal·lat, pot ser utilitzat per qualsevol usuari, donat que no hi ha interacció amb el dispositiu. Tot ell ve automatitzat i els elements operen mitjançant el polsador de la cisterna. Per tant, es podria dir que aquesta es l'única interacció que hi ha, la qual es exactament a l'existent.

D'aquesta manera, una vegada realitzada la instal·lació i que els usuaris ja poden fer ús del dispositiu, únicament, ha calgut l'adquisició de les dades. Aquestes han estat adaptades i organitzades per la posterior visualització en el web. Aquest últim caldria dedicar-li un estudi exactitud del interfase i l'experiència del usuari.

Planificació i metodologia

La planificació inicial ha estat , en certa mesura, alterada per certs problemes amb alguns dispositius. Aquestes circumstancies han afectat a altres tasques les quals han estat realitzades sense un temps òptim de dedicació.

A més com que alguns materials s'han comprat a països estrangers ha provocat que la instal·lació del dispositiu sigui més curta afectant al temps necessària per tal de realitzar algunes proves i verificacions.

Per tant, es considera que es òptim una recerca més acurada dels elements que es requereixen, així com intentar instal·lar elements que ja han estat integrats amb Arduino, per tal de realitzar un desenvolupament més ràpid i senzill.

D'aquesta manera s'evita perdre temps i dedicar-lo a altres tasques que requereixen més dedicació. Per exemple, es considera que calia haver pensat ,inicialment, la instal·lació del mòdul Wireless, per tal d'evitar la realització de proves amb un element quan el desenvolupament està pròxim a la finalització. Així com la posterior modificació d'aquest element per aplicar la nova implantació.

El Futur

Amb aquestes observacions, hauria estat prou necessari la implementació d'una bateria per tal de poder instal·lar el dispositiu en qualsevol instal·lació, sense requerir d'una toma elèctrica. Aquest punt ha estat treballat, però com que no s'han realitzat les proves necessàries s'ha desestimat.

En addició, per tal de reduir el consum d'energia necessari per fer funcionar Arduino, era necessari la implementació d'alguna de les llibreries *Sleep*, les quals aturen els processos del microcontrolador i, únicament, activar-los quan algun esdeveniment succeeix.

Aquest últim, també ha estat estudiat, però certs problemes de codificació amb la el microcontrolador i la falta de temps per estudiar les diferents circumstancia, han condicionat la no implementació d'aquestes llibreries en el desenvolupament del dispositiu.

En línies generals el dispositiu és operatiu, així que a partir d'aquí el producte pot començar el seu camí i poder implementar-se o adaptar-se amb els diferents requeriments que pot tenir cada instal·lació.

6. Glossari

AJAX

Tècnica de desenvolupament web per crear aplicacions interactives.

Analògic

Es refereix a les magnituds o valors que varien amb el temps en forma contínua i poden representar-se en forma d'ones.

Arduino

Arduino és una plataforma de prototips electrònica de codi obert (*open-source*) basada en maquinari i programari flexibles i fàcils d'utilitzar.

Atmel

És una empresa que fabrica els microprocessadors, majoritàriament, utilitzats en les plaques Arduino.

C, C++

Llenguatge de programació.

Chart.js

Aplicació en *Javascript* per la creació de gràfiques.

Circuit

Conjunt de conductors que són correguts per corrent elèctric, i on es troben intercalats aparells productors o consumidors de corrent.

Digital

Dispositiu o mètode que utilitza variacions discretes en voltatge, freqüència, amplitud, ubicació, entre d'altres; per xifrar, processar o transportar senyals binaris (0 o 1) per a dades informàtiques, so, vídeo o una altra informació.

Dispositiu

Mecanisme d'un aparell o equip que, una vegada accionat, desenvolupa de forma automàtica la funció que té assignada.

Firmware

És un bloc d'instruccions de programa per a propòsits específics, gravat en una memòria de tipus no volàtil (ROM, EEPROM, flaix), que estableix la lògica de més baix nivell que controla els circuits electrònics d'un dispositiu de qualsevol tipus.

Font d'Alimentació

Unitat que subministra energia elèctrica a un altre component d'una màquina.

FTDI

Empresa que fabrica els populars xips USB to serial utilitzats en molts Arduinos i accessoris.

HTML

Acrònim de HyperText Markup Language, és un llenguatge de marcat que s'utilitza per al desenvolupament de pàgines d'Internet.

IDE

Entorn de Desenvolupament Integrat, és un programa informàtic compost per un conjunt d'eines integrades que faciliten la programació.

Javascript

Llenguatge de programació d'alt nivell orientat a objectes.

Microcontrolador

Un microcontrolador és un circuit integrat programable que conté tots els components d'un computador, s'empra per realitzar una tasca determinada per la qual ha estat programat. Disposa de processador, memòria per al programa i les dades, línies d'entrada i sortida de dades i sol estar associat a múltiples recursos auxiliars. Pot controlar qualsevol cosa i sol estar inclòs en el mateix dispositiu que controla.

PHP

Acrònim de Hypertext Preprocessor, és un llenguatge de codi obert molt popular especialment adequat per al desenvolupament web i que pot ser incrustat en HTML.

PWM

Modulació per amplada de polsos, consisteix a modificar el cicle d'un senyal periòdic.

RAM

Acronim de Random Access Memory. Una memòria RAM o d'accés aleatori s'utilitza freqüentment en informàtica per a l'emmagatzematge de programes i dades informatives.

Relé

Dispositiu electromagnètic que, estimulat per un corrent elèctric molt feble, obre o tanca un circuit en el qual es dissipa una potència molt major que en el circuit estimulator.

Resistència

És l'oposició que ofereix un material al pas dels electrons (el corrent elèctric). Quan el material té molts electrons lliures, com és el cas dels metalls, permet el pas dels electrons amb facilitat i se li crida conductor.

ROM

Acrònim de Read Only Memory. Una memòria ROM és aquella memòria d'emmagatzematge que permet només la lectura de la informació i no la seva destrucció, independentment de la presència o no d'una font d'energia que l'alimenti.

RX

Abreviació de recepció en telecomunicacions.

Sèrie

Protocol que envia les dades a través d'un filferro un bit darrere l'altre.

Sistema

Conjunt organitzat d'elements diferenciats la interrelació dels quals i interacció suposa una funció global.

Transistor

És un dispositiu electrònic semiconductor que s'utilitza com a amplificador o commutador electrònic.

TX

Abreviació de transmissió en telecomunicacions.

USB

Universal Serial Bus, protocol de connexió que permet enllaçar diversos perifèrics a un dispositiu electrònic per a l'intercanvi de dades, el desenvolupament d'operacions i, en alguns casos, la càrrega de la bateria del dispositiu o dispositius connectats.

Wireless, WIFI

Terme usat per descriure les telecomunicacions en les quals les ones electromagnètiques, en comptes de cables, porten la senyal sobre part o tota la trajectòria de la comunicació.

7. Bibliografia

Llibre consultats:

Banzi, Maximo. *Introducción a Arduino*. [En papel]. Madrid: Ediciones Anaya Multimedia (Grupo Anaya S.A.), 2016. 208 p. ISBN: 9788441537446.

Llocs web de referència:

[1] **Por qué se está acabando el agua**. BBC Mundo, 2015 [consulta octubre 2016].

http://www.bbc.com/mundo/noticias/2014/08/140821_tierra_agua_escasez_finde_dv

[2] **AT6**. Sidar website, 2016. [consulta octubre 2016]. <http://www.sidar.es/depuradora-de-oxidacion-total-aguas-residuales-modelo-at6/>

[3] **ATCP12**. Sidar website, 2016. [consulta octubre 2016]. <http://www.sidar.es/estacion-depuradora-de-aguas-residuales-atcp12/>

[4] **AQUS**. Ecohoe website, 2016. [consulta octubre 2016]. <http://ecohoe.com/>

[5] **600 Series**. Basement Pumping website, 2016. [consulta octubre 2016].

http://www.drainage-systems-online.co.uk/acatalog/Basement_Drainage_System.html?gclid=CjwKEAiAtefDBRDTnbDnvM735xISJABlvGOv3MiddMtYONmhaflaJ7Q49qJZJpCSSB4e41C2xyJn5xoCwafw_wcB#a137

[6] **GW210**. Aquaco website, 2016. [consulta octubre 2016].

<http://www.aquaco.co.uk/images/Aquawiser-GW210-Indirect-Grey-Water-Datasheet.pdf>

[7] **Hogares Verdes**. Ministeri de Agricultura, Alimentació y Medi Ambient, 2009 [consulta novembre 2016]. http://www.magrama.gob.es/es/ceneam/programas-de-educacion-ambiental/hogares-verdes/preguntas_hv.aspx

[8] **Pro Micro - 5V/16MHz**. Sparkfun, 2015 [consulta novembre 2016].

<https://www.sparkfun.com/products/12640>

[9] **NodeMCU**. Esp8266, 2016. [consulta novembre 2016].

<http://www.esp8266.com/wiki/doku.php?id=nodemcu>

Arduino. Arduino Website, 2016 [consulta últim trimestre de 2016]. Disponible en:

<http://www.arduino.cc>

Fritzing. Fritzing, 2016 [consulta último trimestre de 2016]. Disponible en <http://www.fritzing.org>

Spartkfun. Spartkfun, 2016 [consulta último trimestre de 2016]. Disponible en <https://www.sparkfun.com/products/13678>

Prometec. Prometec, 2016 [consulta último trimestre de 2016]. Disponible en <http://www.prometec.net/intro-programacion/>

Arduino. Wikipedia, 2016 [consulta último trimestre de 2016]. Disponible en <https://es.wikipedia.org/wiki/Arduino>

NodeMCU. NodeMCU, 2016 [consulta Diciembre de 2016]. Disponible en http://www.nodemcu.com/index_en.html

8. Annexos

Annex I: Codi dels dispositius

A continuació es detalla quines han estat les funcions més importants del codi de cadascun dels elements explicats en el punt 2. referent al desenvolupament.

Codi implementació ESP8266-01

Com s'ha mencionat, aquest dispositiu té l'habilitat de rebre comandes AT a través del port sèrie. A continuació, es presenta un resum de com s'ha establert la connexió. Aquest codi no es trobarà en el codi final, donat que finalment ha quedat descartat.

Primerament, s'ha de definir la llibreria de comunicacions sèrie, la qual està integrada en el conjunt de llibreries d'Arduino.

```
1 #include <SoftwareSerial.h>
2
```

Figura 36. Definició llibreria de comunicacions sèrie.

Seguidament, en el *setup* d'Arduino s'estableix la connexió sèrie amb ESP8266-01, mitjançant les comandes AT.

```
4 void setup() {
5
6     delay(1000);
7
8     Serial1.println("AT+RST");
9     Serial.println("Resetting the module...");
10    delay(1000);
11    if(Serial1.find("ready")){
12        Serial.println("WiFi - Modul conectat");
13    }
14    else{
15        Serial.println("WiFi - Modul no conectat");
16        return;
17    }
18    delay(3000);
19
20    Serial1.println("AT+CWMODE=1");
21
22    String cmd = "AT+CWJAP=\"";
23    cmd+=SSID;
24    cmd+="\", \"";
25    cmd+=PASS;
26    cmd+="\"";
27
28    Serial1.println(cmd);
29
30    delay(5000);
31
32    if(Serial1.find("OK")){
33        Serial.println("Wifi Connectat.");
34        Serial.println();
35    }
36    else{
37        Serial.println("No pot connectar al wifi");
38        Serial.println();
39        return;
40    }
41
42    delay(5000);
43    Serial1.println(F("AT+CIPMUX=0"));
44
45 }
```

Figura 37. Paràmetres de comunicació sèrie amb ESP8266-01

Una vegada realitzada la connexió amb el ESP-01, ja es pot establir la connexió amb la xarxa.

```
49 void informacion()
50 {
51     String cmd = "AT+CIPSTART=\"TCP\", \"";
52
53     cmd += "ip"; // IP//
54     cmd += "\",80"; // PUERTO//
55
56     Serial1.println(cmd);
57
58     if(Serial1.find("Error")) return;
59     cmd = "GET /add_data.php?";
60     cmd += "valors=";
61     cmd += litros;
62     cmd += " HTTP/1.0\r\nHost: IP DEL SERVIDOR\r\n\r\n";
63
64     Serial1.print("AT+CIPSEND=");
65     Serial1.println(cmd.length());
66
67     if(Serial1.find(">")){
68         Serial.println("Connection established.");
69     }
70     else{
71         Serial1.println("AT+CIPCLOSE");
72         Serial.println("Connection timeout.");
73         delay(1000);
74         return;
75     }
76
77     Serial1.println(cmd);
78
79 }
```

Figura 38. Codi de comunicació sèrie amb el ESP-01.

Codi implementació Vàlvula d'entrada

Per tal de reutilitzar el codi per altres dispositius amb característiques semblant s'ha optat per crear un llibreria la qual posteriorment farem ús en el projecte principal.

```
1 #include "Valve.h"
2
3
4 Valve::Valve(int pin)
5 {
6     pinMode(pin, OUTPUT);
7     _pin = pin;
8 }
9
10 void Valve::AbrirSolenoid()
11 {
12
13     Serial.println("Abriendo valvula");
14     digitalWrite(_pin, HIGH);
15 }
16
17
18 void Valve::CerrarSolenoid()
19 {
20
21     Serial.println("Cerrando valvula");
22     digitalWrite(_pin, LOW);
23 }
```

Figura 39. Llibreria de la vàlvula d'entrada

Fent ús de la funció `AbrirSolenoid()` se li indica a la placa Arduino que activi la sortida on està connectat la vàlvula, provocant l'apertura de la vàlvula i la posterior entrada d'aigua en el depòsit.

De manera contrària, amb la funció `CerrarSolenoid()`, Arduino desactiva el pas de corrent provocant que la vàlvula vagi al seu estat inicial, bloquejant així el pas d'aigua cap al depòsit.

Codi implementació Bomba d'aigua

Com s'ha mencionat abans, en aquest també s'ha optat per crear un llibreria amb les funcions que requereixen per, posteriorment en el projecte principal, posar en funcionament la bomba d'aigua.

```
1  #include "WaterPump.h"
2
3
4  WaterPump::WaterPump(int pin)
5  {
6      pinMode(pin, OUTPUT);
7      _pin = pin;
8  }
9
10 void WaterPump::WaterPumpON()
11 {
12
13     Serial.println("Abriendo bomba");
14     digitalWrite(_pin, HIGH);
15
16 }
17
18 void WaterPump::WaterPumpOFF()
19 {
20
21     Serial.println("Cerrando bomba");
22     digitalWrite(_pin, LOW);
23
24 }
```

Figura 40. Llibreria de la Bomba d'aigua

Codi implementació Vàlvula de tres vies

En aquest cas, s'ha fet ús del codi dissenyat per la vàlvula magnètica, mencionada anteriorment. El propòsit era exactament el mateix i la funció és similar. Per tant, únicament, s'ha de trucar a aquesta llibreria e indicar-li a quin pin estan connectats cadascun dels elements que es compona aquesta vàlvula de tres vies.

Codi implementació Sensor de nivell

El codi d'aquest dispositiu és molt simple, donat que l'únic a realitzar es veure quan el circuit, en el pin definit, es tanca o s'obre. Aquest serà el que provoqui l'activació de la nostra funció principal, la qual omplirà la cisterna.

```
2
3 if (digitalRead(2) == 0) { //Comprobar primero si el flotador esta abajo.
```

Figura 41. Codi verificació del estatus pin 2.

Per tant, quan el *if*, tal com indica en la imatge anterior, té valor 0, significa que el circuit està obert i es començarà a activar cadascuna de les funcions i condicions definides dintre d'aquest.

Codi implementació Sensor de flux d'aigua.

Per posar en funcionament el sensor de flux, primerament, s'ha de tenir en compte que haurem de fer ús d'un pin que pugui llegir les interrupcions.

Per tant, en el *Setup*, haurem de indicar quin serà aquest pin e indicar-li, una vegada estigui llegint aquestes interrupcions, quina funció haurà de executar.

```
1 void setup()
2 {
3   /*INTERRUPTS PER ARDUINO MICRO 32u4 based Micro
4      0 = PIN 3
5      1 = PIN 2
6      2 = PIN 0
7      3 = PIN 1
8      4 = PIN 7
9   */
10  pinMode(sensorDeFlujo, INPUT);
11
12
13  attachInterrupt(0, flujo, RISING); // configuracio Interrupt
14
15  interrupts(); // Habilitar interrupciones
16  tiempoAnterior = millis();
17 }
```

Figura 42. Configuració dels *interrups* en la funció *setup()*.

En aquest, la funció que s'executarà és un comptador que comptabilitza el número de polsos o interrupcions que genera el sensor de flux quan el molinet gira.

```
19 void flujo () // Funcion de interrupcion
20 {
21   pulsos++; // Simplemente sumar el numero de pulsos
22 }
```

Figura 43. Definició de la funció comptadora de polsos.

Una vegada realitzada aquesta comptabilització es procedeix a calcular el número de litres que han passat per dintre del sensor flux.

El càlcul en qüestió agafa el número de polsos rebuts de la funció anterior sabent que el número de polsos equivalent a un litre es realitza una regla de tres. El valor de polsos equivalent a un litre venen determinats per les indicacions del fabricant i un calibratge prèviament realitzat sobre el sensor.

```
24 lhora = (pulsos * 60 / 6.67); // (Frecuencia dels pulsos x 60 min) / 7.5Q = ratio flujo en L/hour
25 pulsos = 0; // reinicien el contador Counter
26
27 Serial.print(" Pulsos totales: ");
28 Serial.print(pulsosAcumulados);
29
30 litros = pulsosAcumulados * 1.0 / 400; //Cada 400 pulsos = 1 litro
31 Serial.print("Litros: ");
32 Serial.println(litros);
33 Serial.print("\n");
```

Figura 44. Funció del càlcul de litres basats en els polsos detectats.

Annex II: Explicació desenvolupament interfície d'usuari

Amb les tres eines mencionades en el punt 3.3. (AJAX, JSON i Chart.js), per tal de dibuixar un gràfic, primerament hem de crear un element *canvas* d'HTML5 en la nostra secció del *bodi* del nostre arxiu HTML.

```
<div id="chart-container2">
  <canvas id="mycanvas2"></canvas>
</div>
<div id="chart-container">
  <canvas id="mycanvas"></canvas>
</div>
```

Figura 45. Definició de *canvas* en HTML.

A continuació, em de crear un script per inicialitzar la classe gràfica i ho definim en el codi HTML.

```
<!-- javascript -->
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript" src="js/Chart.min.js"></script>
<script type="text/javascript" src="js/app.js"></script>
```

Figura 46. Definició dels scripts en HTML.

Aquest script començarà realitzant trucades AJAX des del script a un fitxer amb sentències PHP.

```
1
2 $(document).ready(function() {
3   $.ajax({
4     url: "index2.php",
5     method: "GET",
6     success: function(data) {
7       console.log(data);
8       var fecha = [];
9       var valor = [];
```

Figura 47. Trucades AJAX a sentències PHP.

Aquestes sentències connecta amb la base de dades i fa una *SELECT* per tal de recuperar els registres inserits, prèviament, per Arduino. El codi PHP recupera i mostra la informació en format JSON.

```

2  <?php
3  //configuración cabecera a json
4  header('Content-Type: application/json');
5
6  //base de dades
7  define('DB_HOST', );
8  define('DB_USERNAME', ');
9  define('DB_PASSWORD', ');
10 define('DB_NAME', ');
11
12 //Connexio
13 $mysqli = new mysqli(DB_HOST, DB_USERNAME, DB_PASSWORD, DB_NAME);
14
15 if(!$mysqli){
16     die("Connection failed: " . $mysqli->error);
17 }
18
19 //query per aconseguir dades de la base de dades
20 $query = sprintf( );
21
22 //executa la query
23 $result = $mysqli->query($query);
24
25 //bucle retorna les dades
26 $data = array();
27 foreach ($result as $row) {
28     $data[] = $row;
29 }
30
31 //libera memoria
32 $result->close();
33
34 //tanca la connexio
35 $mysqli->close();
36
37 //ahora imprimeix les dades en json
38 print json_encode($data);
39 -?>

```

Figura 48. Codi PHP de connexió i consulta amb la BD per la posterior impressió en JSON.

Aquestes dades són llegides pel script, el qual anirà generant el gràfic amb aquesta informació.

Amb aquest script també es pot definir la configuració gràfica que tindrà el gràfic, com el color de les barres, línies, el color de fons de la gràfica o etiquetes, entre d'altres.


```

var chartdata = {
  labels: dia,
  datasets : [
    {
      label: '#Litres estalviats per dia durant aquest mes',
      lineTension: 0.1,
      backgroundColor: "rgba(75,192,192,0.4)",
      borderColor: "rgba(75,192,192,1)",
      borderCapStyle: 'butt',
      borderDash: [],
      borderDashOffset: 0.0,
      borderJoinStyle: 'miter',

      data: valores
    }
  ]
};

var ctx = $("#mycanvas");

var barGraph = new Chart(ctx, {
  type: 'line',
  data: chartdata
});

```

Figura 49. Configuració dels gràfics en script chart.js

Per últim, tal com es defineix en la imatge anterior, l'últim que s'hauria de realitzar per col·locar el gràfic en la web, seria indicar en quin element *canvas* es vol incrustar la gràfica.

Annex III: Instal·lació Servidor Web per realització

Per tal d'implementar l'aplicació web cal tenir un servidor on s'enviaran les dades i a on s'accedirà per tal de consultar aquesta informació. Aquest servidor ha de tenir, com a mínim accessibles, les aplicacions menciones en el punt 3.2. i 3.3.

En aquest cas s'ha fet us de l'aplicació WAMPServer. Aquest es un entorn de desenvolupament web que ens permet tenir el nostre propi servidor o host local (instal·lat en el nostre ordinador).

Per tal de realitzar la instal·lació:

1. El primer que farem serà descarregar-nos Visual C++ Redistributable per a Visual Studio i Paquets redistribuïbles de Visual C++ per a Visual Studio.
2. S'ha de descarregar des de la seva pròpia web <http://www.wampserver.com>.
3. Una vegada descarregat es procedeix a executar el instal·lador i ens apareixerà la selecció de idiomes. On s'ha de fer clic sobre el OK.

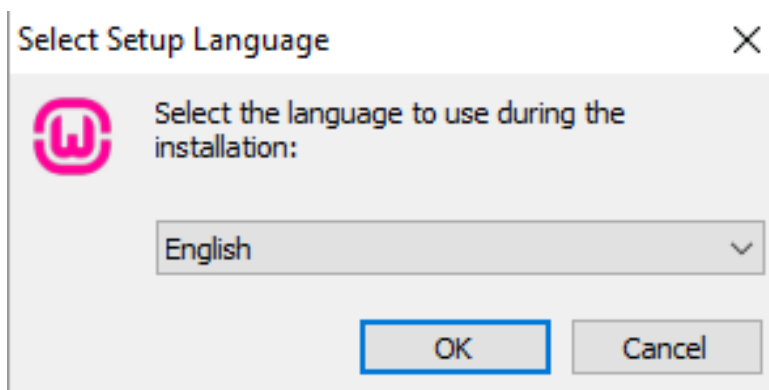


Figura 50. Selecció de idioma en el instal·lador de wamp.

4. S'accepten els termes i condicions per seguir amb la instal·lació.

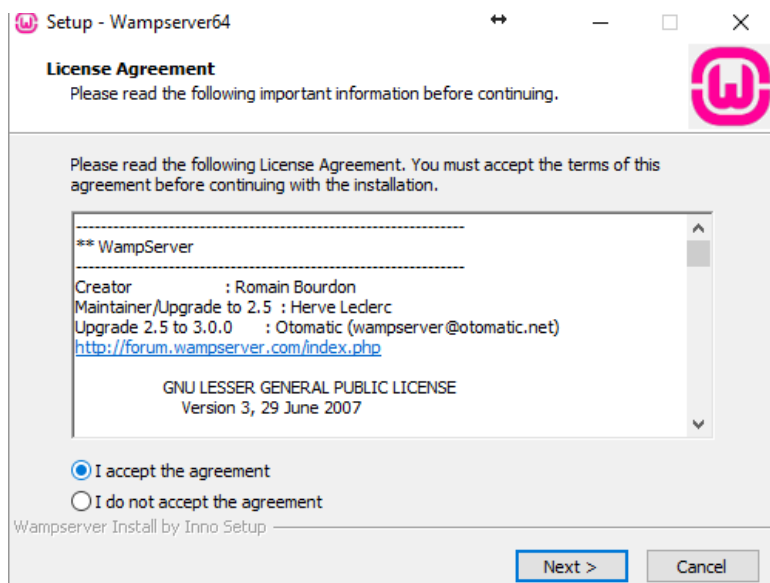


Figura 51. Acceptació dels termes i condicions.

5. A continuació apareixerà una nota informativa. Una vegada llegida s'ha de fer click en "NEXT".

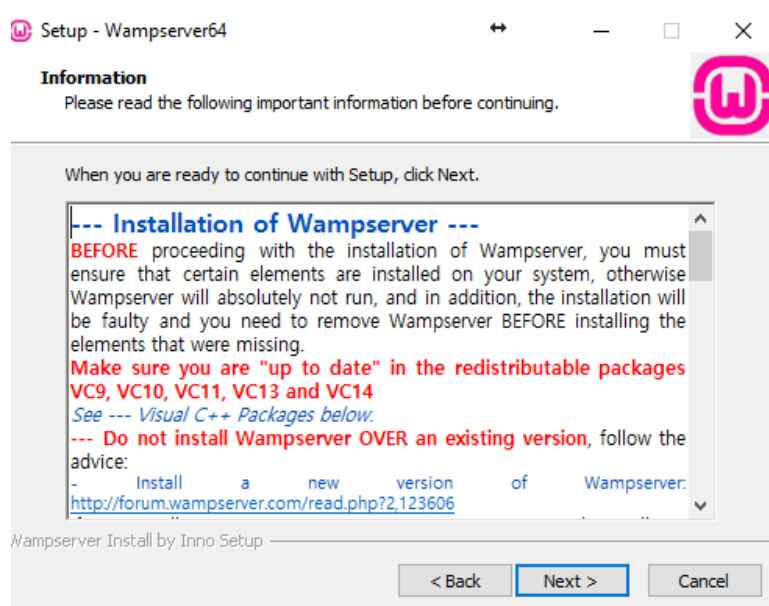


Figura 52. Nota informativa

6. Es selecciona la carpeta de destí de la instal·lació. Si es deixa per defecte s'instal·larà en la arrel del disc principal. Una vegada indica es fa click en el botó "NEXT".

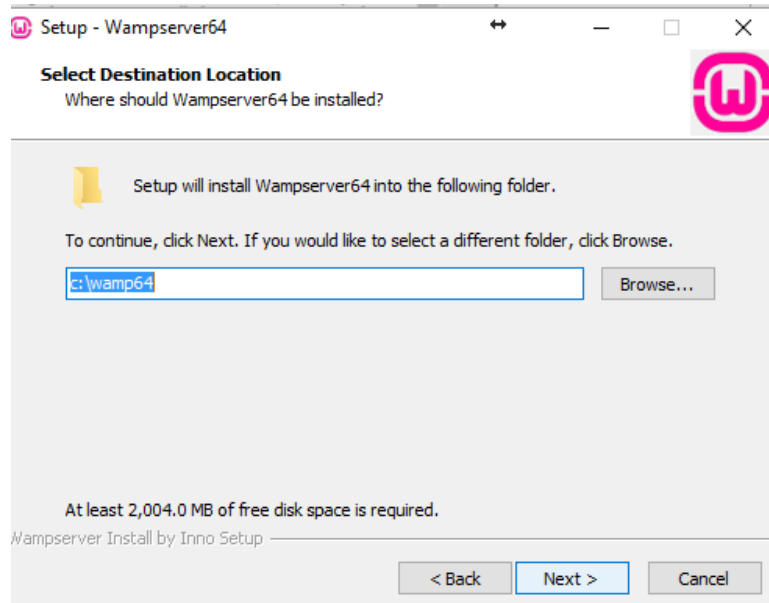


Figura 53. Carpeta de destí de la instal·lació

7. En el següent pas s'haurà de indicar on es crearà l'accés directe en el menú d'inici. Una vegada seleccionada cal fer click en "NEXT".

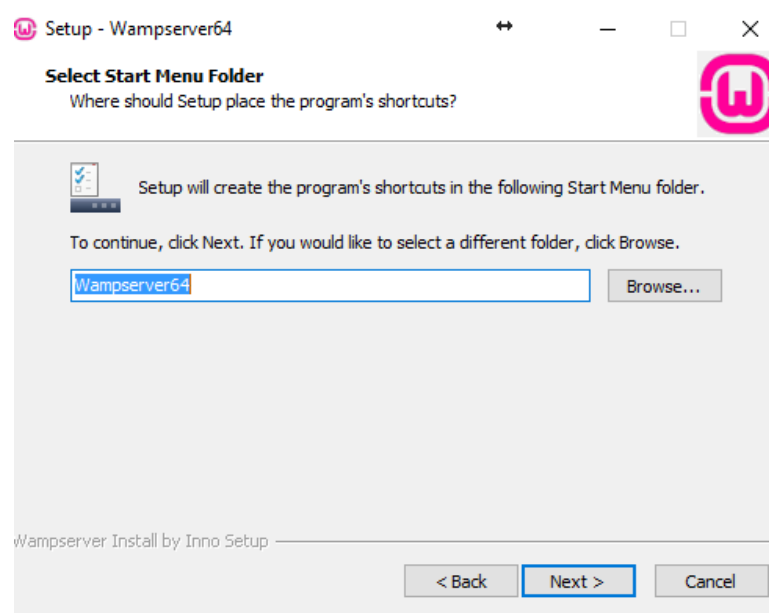


Figura 54. Accés directe en el menú d'inici.

8. En el següent procés s'observa el sumari de instal·lació. Si està tot correcte es procedeix a fer click en el botó "INSTALL".

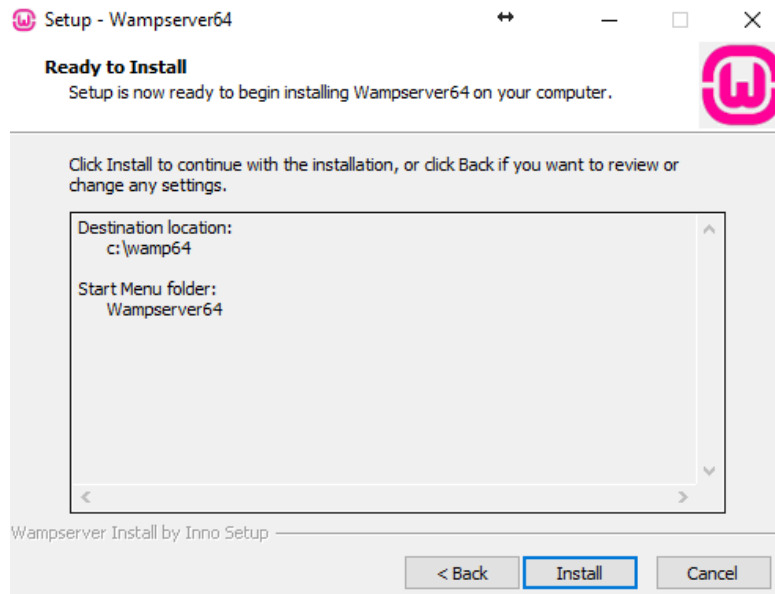


Figura 55. Sumari de instal·lació.

9. El procés de instal·lació inicia la seva tasca

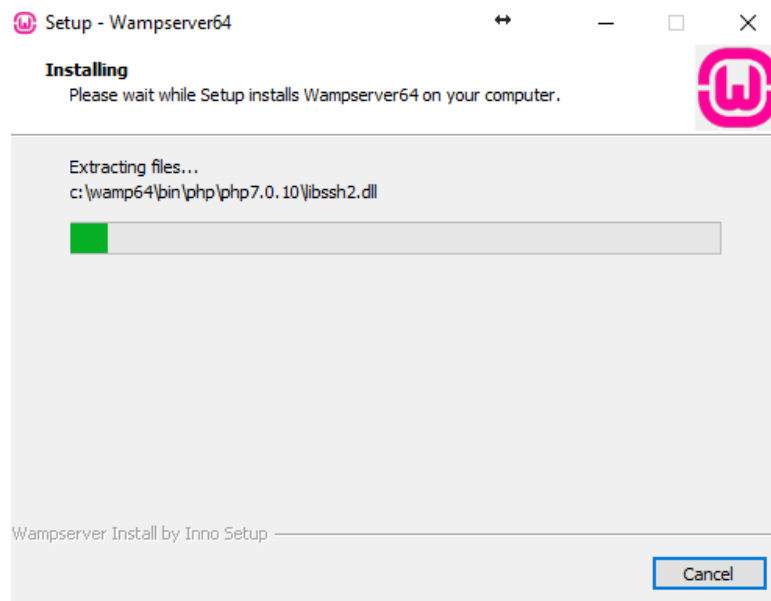


Figura 56. Procés de instal·lació.

10. En aquesta finestra se indicarà el navegador web que WampServer farà servir com a predeterminar .

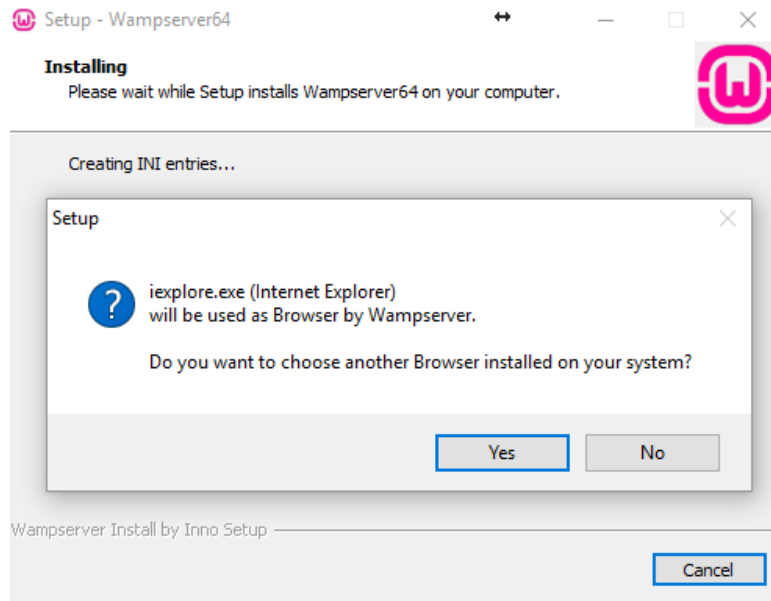


Figura 57. Selecció navegador web

11. Apareixerà una nota informativa. Una vegada llegida cal fer clic en el botó “NEXT”.

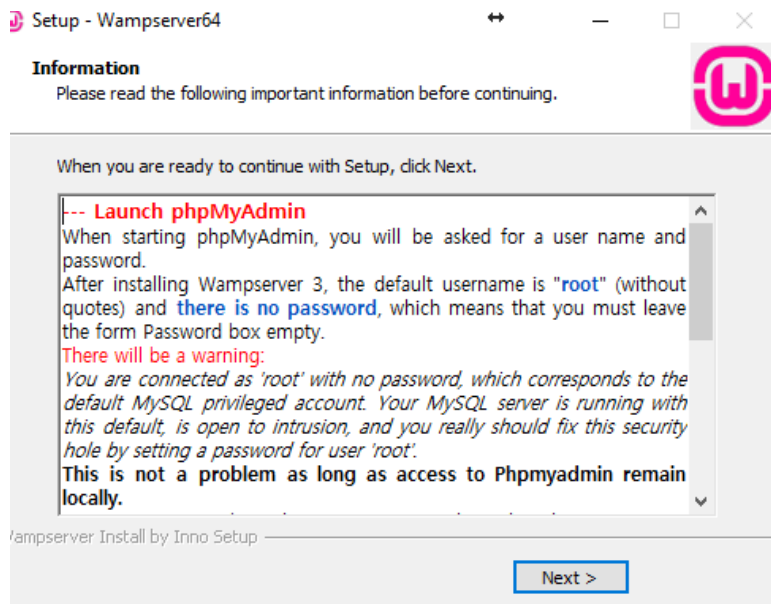


Figura 58. Nota informativa final de instal·lació

12. El procés de instal·lació ha finalitzat, i ja podem executar l'aplicació deixant marcat la opció llançament i donar a finalitzar.

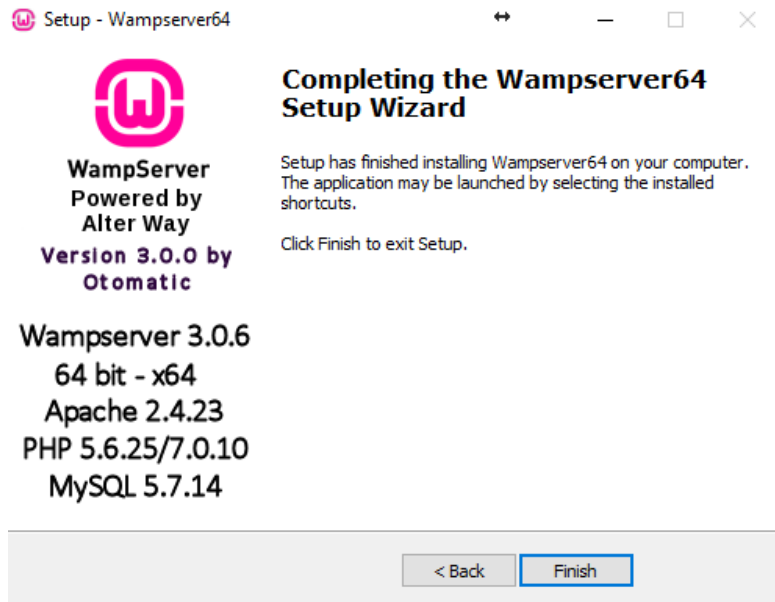


Figura 59. Finestra de instal·lació completa.

13. WampServer se executarà en segon pla en la barra de tasques i es tornarà d'un color verd.

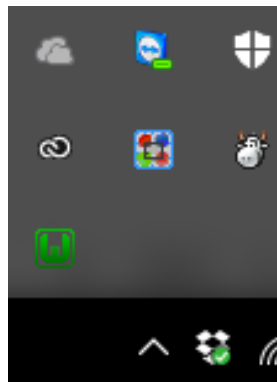


Figura 60. Barra de tasques amb wamp iniciat.

Annex IV: Creació de la base de dades

Una vegada realitzada la instal·lació de WAMP en el punt anterior i, realitzada la posterior execució de l'aplicació, es procedeix a accedir via web a l'aplicació PHPmyAdmin:

<http://localhost/phpmyadmin>

Una vegada dintre de l'aplicació, el procés per realitzar la instal·lació de la base de dades es:

1. Fer clic sobre la pestanya “Importar”:

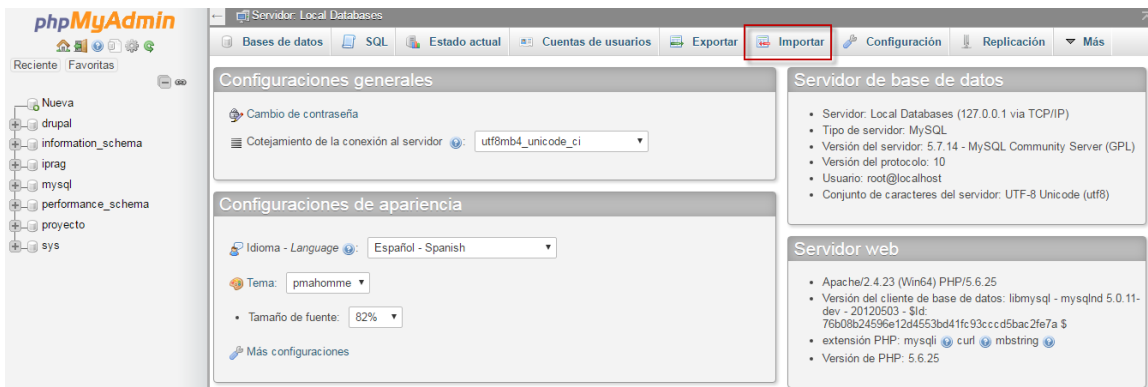


Figura 61. Finestra de inici phpMyAdmin.

2. PHP mostrarà la finestra de importació. En aquesta finestra, s'ha de fer clic sobre el botó “Choose File”:

Bases de datos SQL Estado actual Cuentas de usuarios Exportar Importar Configuración Replicación Más

Importando al servidor actual

Archivo a importar:

El archivo puede ser comprimido (gzip, bzip2, zip) o descomprimido.
Un archivo comprimido tiene que terminar en **[formato].[compresión]**. Por ejemplo: **.sql.zip**

Buscar en su ordenador: **Choose File** No file chosen (Máximo: 128MB)

También puede arrastrar un archivo en cualquier página.

Conjunto de caracteres del archivo: utf-8

Importación parcial:

Permitir la interrupción de una importación en caso que el script detecte que se ha acercado al límite de tiempo PHP. (Esto podría ser un buen método para importar archivos grandes; sin embargo, puede dañar las transacciones.)

Omitir esta cantidad de consultas (en SQL) desde la primera: 0

Otras opciones:

Habilite la revisión de las claves foráneas

Formato:

SQL

Figura 62. Finestra de importació phpMyAdmin.

3. En la nova finestra del explorador, s'ha de buscar i seleccionar el script SQL, el qual crearà la base de dades.

El fitxer **Proyecto.sql es troba junt a aquest document.*

4. Quan el fitxer ha estat seleccionat, l'aplicació torna a la finestra de importació. Desplaçar-se cap al final d'aquesta pàgina, es trobarà el botó "Continuar" per tal d'executar el script.

Opciones específicas al formato:

Modalidad SQL compatible: NONE

No utilizar AUTO_INCREMENT con el valor 0

Continuar

Figura 63. Botó Continuar en la finestra de importació phpMyAdmin.

Annex V: Implementació de l'aplicació web.

Tenint en compte que la instal·lació del servidor web, mencionada en el Annex III i que la creació de la base de dades, indicada en el Annex IV, han estat realitzades, es procedeix a instal·lar l'aplicació web de la següent manera:

1. Anar a la carpeta on s'ha instal·lat WAMP. Si s'ha indicat la carpeta per defecte, aquesta seria c:/wamp o c:/wamp64.
2. Dintre de la carpeta principal de WAMP es troba la carpeta "www".

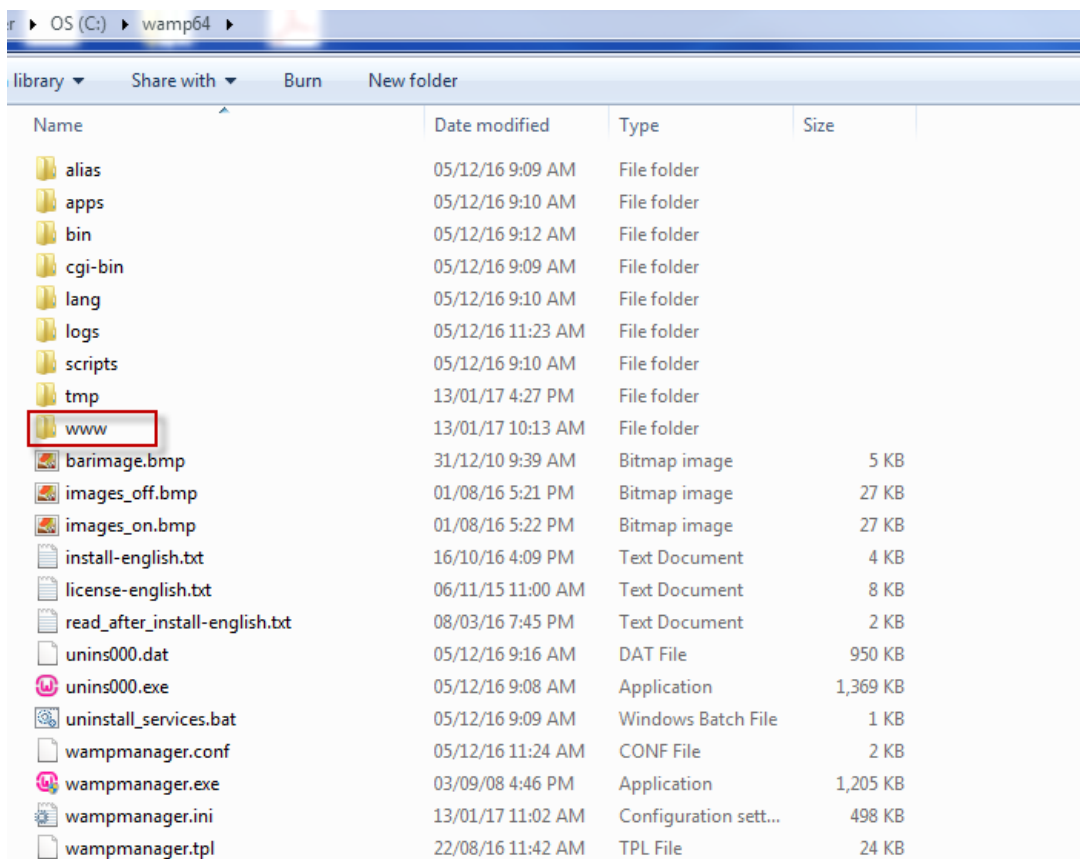


Figura 64. Carpeta arrel de wamp.

3. Dintre d'aquesta carpeta es copia tots el fitxers referent a l'aplicació web.

**Aquest fitxers es troben junt a aquest document.*

Name	Date modified	Type	Size
js	14/01/17 11:35 AM	File folder	
add_data.php	30/12/16 2:53 PM	PHP File	1 KB
cabecera.css	29/12/16 12:26 PM	Cascading Style S...	2 KB
config.php	29/12/16 4:49 PM	PHP File	1 KB
EnCons.html	29/12/16 2:05 PM	Chrome HTML Do...	1 KB
index2.php	30/12/16 3:55 PM	PHP File	1 KB
iprag.css	30/12/16 4:55 PM	Cascading Style S...	2 KB
login.css	29/12/16 12:22 PM	Cascading Style S...	2 KB
login.php	30/12/16 5:02 PM	PHP File	2 KB
logout.php	07/12/16 7:41 PM	PHP File	1 KB
review_data.php	25/12/16 9:07 PM	PHP File	2 KB
summari.php	29/12/16 7:22 PM	PHP File	1 KB
summari2.php	29/12/16 7:33 PM	PHP File	1 KB
welcome.php	30/12/16 4:56 PM	PHP File	3 KB

Figura 65. Fitxer de l'aplicació web.

Annex VI: Valors obtinguts durant el període de prova.

La següent taula detalla l'estalvi en cadascú del moments on s'acciona la descarrega d'aigua de la cisterna durant el període de proves realitzat.

Data	Litres
2016-12-18 03:52:36	9.78
2016-12-18 05:02:23	9.69
2016-12-18 11:30:19	4.72
2016-12-18 17:21:07	4.5
2016-12-18 19:21:24	8.2
2016-12-18 21:39:48	0.72
2016-12-18 23:17:51	0.11
2016-12-19 02:58:06	4.3
2016-12-19 07:43:12	5.2
2016-12-19 08:00:00	9.68
2016-12-19 08:30:00	9.7
2016-12-19 16:25:09	0
2016-12-19 20:13:04	0
2016-12-19 22:11:59	4.2
2016-12-19 23:19:44	0
2016-12-20 05:02:48	3.8
2016-12-20 07:47:24	2.1
2016-12-20 12:26:15	1.3
2016-12-20 18:24:16	3.2
2016-12-20 19:21:00	0
2016-12-20 21:04:26	4.2
2016-12-21 02:11:48	0
2016-12-21 05:08:24	5.2
2016-12-21 07:46:37	0.22
2016-12-21 18:16:28	0
2016-12-21 21:18:07	6.57
2016-12-21 22:45:04	3.12
2016-12-22 01:28:19	0
2016-12-22 05:02:13	0
2016-12-22 07:47:05	4.44
2016-12-22 17:28:11	2.12
2016-12-22 19:21:06	0
2016-12-22 23:27:31	0
2016-12-23 05:03:12	5.23
2016-12-23 21:19:36	0
2016-12-23 22:27:49	0
2016-12-24 10:58:08	6.83
2016-12-24 18:57:43	0

2016-12-24 23:12:47	0
2016-12-24 23:45:17	5.12
2016-12-25 10:00:15	9.72
2016-12-25 12:00:26	9.7
2016-12-25 13:07:01	5.1
2016-12-25 17:07:07	0.2
2016-12-25 20:07:14	0.12
2016-12-25 21:07:20	0.5
2016-12-26 08:00:15	2.72
2016-12-26 08:30:26	0.7
2016-12-26 13:07:01	0.1
2016-12-26 18:07:07	0.5
2016-12-26 21:07:14	0
2016-12-26 21:33:20	0.5
2016-12-27 08:28:26	0.2
2016-12-27 13:27:03	0
2016-12-27 18:17:17	0
2016-12-27 21:17:11	0
2016-12-27 21:33:20	0.12
2016-12-28 14:57:53	4.8
2016-12-28 18:37:37	1.2
2016-12-28 21:17:11	2.1
2016-12-28 21:24:11	9.71
2016-12-28 21:33:20	0.12
2016-12-28 21:33:40	1.42
2016-12-29 08:40:44	1.2
Total	174.98

Taula 6. Estalvi en cada moment de la prova.

A partir d'aquests resultats s'ha pogut realitzar els càlculs i resultats finals per tal de determinar les conclusions mencionades, anteriorment, en la memòria.

