

IES POLITÉCNICO HERMENEGILDO LANZ

DEPARTAMENTO DE INFORMÁTICA



RIEGO AUTOMATICO

PROYECTO FINAL DEL GRADO SUPERIOR DE DESARROLLO DE
APLICACIONES WEB

PROYECTO REALIZADO POR JAIME NAVARRO LÓPEZ

2 DE JUNIO DE 2020

ÍNDICE

RESUMEN	2
1. INTRODUCCIÓN	3
1.1 METODOLOGÍA Y CASOS DE USO	3
2. DISEÑO DEL SITIO	4
2.1 ESTRUCTURA DE LA WEB	4
2.2 PLANIFICACIÓN DE LA WEB	4
2.3 ESTILO DE LA WEB	12
3. CONFIGURACIÓN DEL SERVIDOR	13
3.1 ESTRUCTURA DE LA BASE DE DATOS	14
4. CONFIGURACIÓN DEL ARDUINO	15
4.1 COMPONENTES USADOS	16
5. BIBLIOGRAFÍA	16

RESUMEN

Este es el documento de memoria para el proyecto final del grado superior de desarrollo de aplicaciones web. En él se detallan mediante distintas secciones desde la fase de planificación hasta la puesta en marcha.

Este proyecto centra su esfuerzo en la creación de una herramienta empezando desde el análisis y planificación hasta la etapa de implementación y puesta en marcha.

El proyecto se ha realizado mediante dos lenguajes de programación, PHP y C++, han sido trabajados mediante dos entornos de desarrollo Visual Studio Code y Arduino. Para el desarrollo de la web se ha usado el framework¹ Symfony.

El nombre de la herramienta es “Riego Fácil”.

Dicha herramienta ayuda a mantener un control de riego mas sencillo a los agricultores y así poder optimizar un recurso tan preciado en este sector como es el agua.

El proyecto está compuesto por un servidor y un despliegue de elementos controlados por un microcontrolador Arduino, que a su vez se comunica con el servidor.

¹ Un framework es una herramienta que te permite automatizar el desarrollo de una aplicación.

1. INTRODUCCIÓN

Un proyecto de fin de estudios se justifica por sí solo, ya que constituye la parte más importante, sintetizar todo lo aprendido durante el curso para poder convertir en un producto tangible una idea. En este caso, el producto es una herramienta que se ha estudiado, planificado, analizado y diseñado para su posterior implantación.

El proyecto surge a partir de la necesidad que tienen los agricultores de regadío, para mantener un control de los riegos que realizan, puesto que el agua es un recurso muy importante y es necesario optimizar el riego en los cultivos.

El producto consta de dos partes fundamentales, un servidor (donde se encuentra alojada la web y la base de datos) y un microcontrolador (que se encarga de activar y desactivar los distintos elementos según estén definidos en la base de datos).

El objetivo principal es la optimización del agua en el proceso de riego y obtener datos de éste para que el agricultor pueda analizar los periodos de riego. Además, también se pretende optimizar los desplazamientos del agricultor a la finca de regadío al poder controlar el riego a distancia mediante la web que se ha desarrollado.

1.1. Metodología y casos de uso

La metodología que se ha seguido para la implementación ha sido un ciclo de vida en cascada, dicho ciclo está compuesto por 6 etapas.

Planificación y análisis, diseño, codificación, pruebas, despliegue y verificación.

Los distintos casos de uso para el cliente que se han desarrollado en esta herramienta son:

- Visualizar los registros de los riegos que se han ejecutado
- Configurar el horario en el que se va a regar cada sector
- Activar y desactivar el riego programado
- Activar y desactivar el riego en tiempo real
- Solicitar nuevos horarios
- Reportar una incidencia

2. DISEÑO DEL SITIO

2.1. Estructura de la web

La web consta en una estructura jerárquica, la vista principal desde la que partimos es la que vemos una vez nos hayamos autenticado, desde esta vista tenemos disponibles realizar distintas acciones, las cuales están todas al mismo nivel, únicamente solo hay una vista que va a colgar solamente de una de estas vistas.

VISTA PRINCIPAL

VER HISTORIAL DE RIEGOS VER ESTADO DE RIEGO VER CONFIGURACIÓN DE RIEGO SOLICITAR HORARIO REPORTAR INCIDENCIA CERRAR SESIÓN

EDITAR SECTOR

La estructura se planificó siguiendo las distintas funcionalidades que iba a tener la herramienta. A continuación, están los esquemas que se han seguido para elaborar las distintas vistas de la aplicación web.

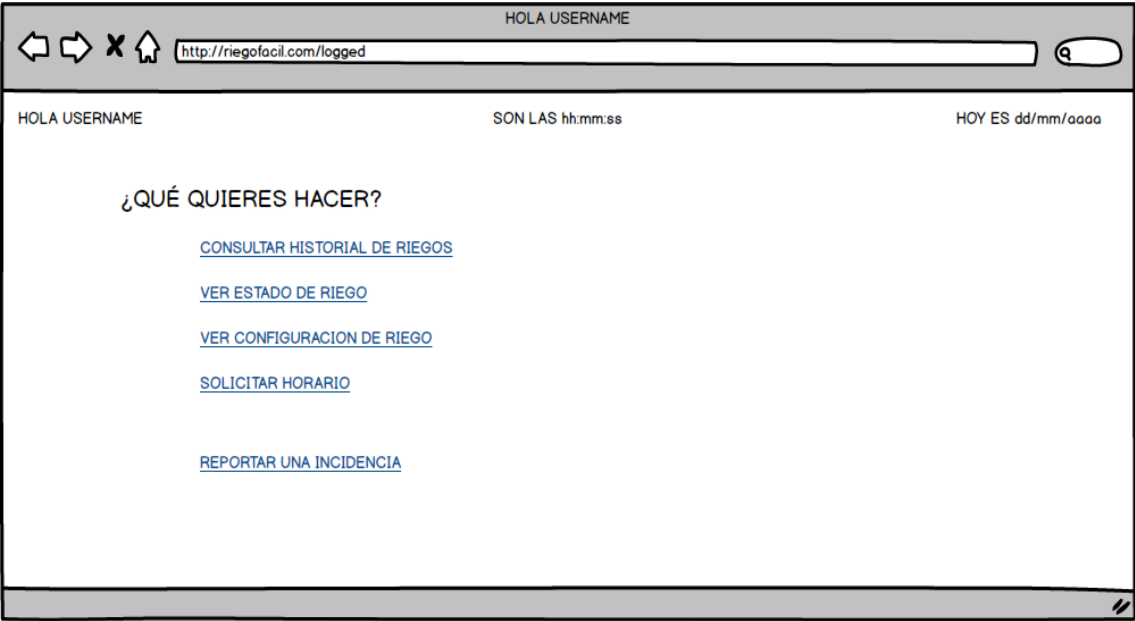
Dicha estructura está basada en el Modelo-Vista-Controlador, con unos controladores claramente diferenciados por cada entidad. Siendo el “MainController” el único en albergar distintas funcionalidades como mostrar la vista inicial, la vista una vez se haya iniciada sesión, la vista para reportar incidencias y el procesamiento del formulario de contacto.

2.2. Planificación de la web

La web, fue inicialmente planificada mediante el programa Balsamiq Mockup, dicho programa te permite realizar pequeños guiones de lo que va a ser la web. Dicha planificación no difiere mucho del resultado final.

A continuación, hay una comparación entre la idea inicial y el diseño final.

Vista principal.



¿QUE QUIERES HACER?

- Ver historial de riegos
- Ver estado de riego
- Ver configuración de riego
- Solicitar horario
- Reportar incidencia
- Cerrar sesión

Vista historial de riegos

HISTORIAL DE RIEGOS

http://riegofacil.com/history

SON LAS hh:mm:ss

HISTORIAL DE RIEGOS

HOY ES dd/mm/aaaa

VOLVER ATRÁS

SELECCIONAR SECTOR

0-TODOS
1-TOMATES
2-HABAS
3-PIMIENTOS
4-BERENJENAS

REINICIAR FILTROS

FILTRAR POR FECHA

INICIO

FINAL

//

//

DIA	SECTOR	HORA INICIO	HORA FINAL	HUMEDAD INICIO	HUMEDAD FINAL	TOTAL LITROS	...
-----	--------	-------------	------------	----------------	---------------	--------------	-----

Inicio

Ver estado de riego

Ver configuracion de riego

Solicitar horario

Reportar incidencia

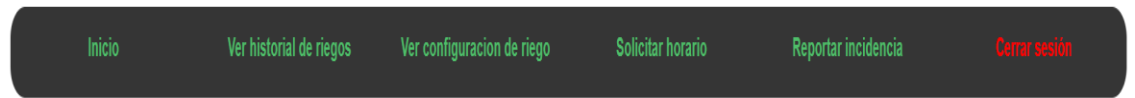
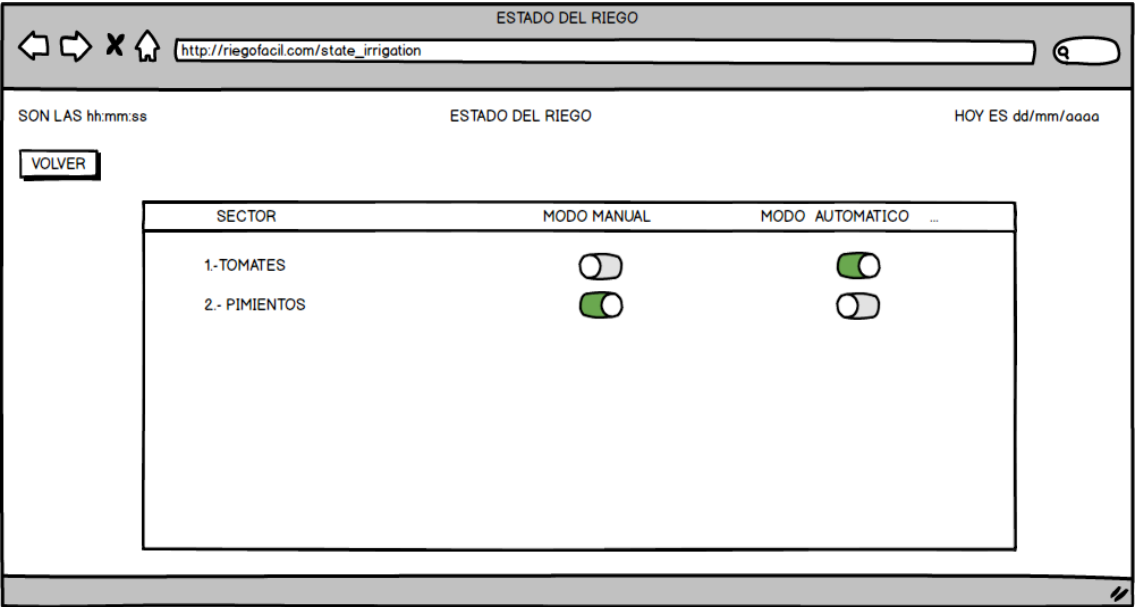
Cerrar sesión

Historial de riegos

Nombre de sector	Fecha	Hora inicio	Hora final	Humedad inicial	Humedad final	Total litros
Tomates	24-05-2020	18:24	18:30	10	75	80
Tomates	24-05-2020	18:24	18:30	10	75	80
Tomates	01-06-2020	19:40	19:45	5	10	20
Tomates	01-06-2020	19:40	19:45	5	10	20
Tomates	01-06-2020	19:40	19:45	5	10	21
Tomates	02-06-2020	16:30	16:31	2	4	1
Tomates	02-06-2020	17:57	17:57	1	3	0
Tomates	02-06-2020	17:57	17:57	1	3	0
Tomates	02-06-2020	18:54	18:54	6	5	0
Tomates	02-06-2020	18:54	18:54	6	5	0
Tomates	02-06-2020	20:44	20:46	4	5	0
Tomates	02-06-2020	20:44	20:46	4	5	0
Tomates	02-06-2020	21:14	21:31	3	6	0
Tomates	02-06-2020	21:14	21:31	3	6	0

6

Vista estado del riego.



Estado del riego

Sector	Encender/Apagar	Programado
Tomates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Vista configuración de riego.

RIEGO

http://riegofacil.com/show_irrigation

SON LAS hh:mm:ss

CONFIGURAR RIEGO

HOY ES dd/mm/aaaa

VOLVER

SECTOR	HORA INICIO	HORA FINAL	REPETICIONES	ACCIONES
				EDITAR

Inicio

Ver historial de riego

Ver estado del riego

Solicitar horario

Reportar incidencia

Cerrar sesión

Estado del riego

Nº Sector	Nombre del sector	Horario	Acciones
1	Tomates	Martes 23.05-23.10	Editar sector

Vista editar sector.

EDITAR RIEGO

http://riegofacil.com/{id_sector}/edit

SON LAS hh:mm:ss

CONFIGURAR RIEGO

HOY ES dd/mm/aaaa

CANCELAR

SECTOR {name_sector}

GUARDAR

MODO ACTUAL : {sector_mode}

☐ MODO 1

☐ MODO 2

☐ MODO 3

☐ MODO 4

INFORMACION DEL MODO SELECCIONADO

Editar Sector Tomates

Nombre

Tomates

Horario

Martes 23:05-23:10

Guardar cambios

Cancelar

Vista solicitar horario.

SOLICITAR MODO

SON LAS hh:mm:ss SOLICITAR NUEVO MODO HOY ES dd/mm/aaaa

CANCELAR ENVIAR

SELECCIONAR DIAS :

☐ LUNES ☐ MARTES ☐ MIERCOLES ☐ JUEVES ☐ VIERNES ☐ SABADO ☐ DOMINGO

HORA INICIO HORA FINAL

HORA INICIO HORA FINAL

INFORMACION ADICIONAL

Solicitar horario

Nombre

Descripcion

Hora de inicio por la mañana

Hora de fin por la mañana

Hora de inicio por la tarde

Hora de fin por la tarde

Lunes ☐

Martes ☐

Miercoles ☐

Jueves ☐

Viernes ☐

Sabado ☐

Domingo ☐

Enviar

Cancelar

Vista reportar incidencia.

ESTADO DEL RIEGO

http://riegofacil.com/state_irrigation

SON LAS hh:mm:ss

REPORTAR INCIDENCIA

HOY ES dd/mm/aaaa

VOLVER

ENVIAR

ASUNTO

CUERPO

Descripcion del problema

Enviar

Cancelar

La navegación por las distintas vistas es bastante fluida ya que en cada vista en la parte superior existe un menú de navegación. Todas las vistas son sencillas e intuitivas, pero para aclarar los distintos casos de uso y la vista a la que corresponde aquí va una pequeña explicación siguiendo el orden de las vistas que hay sobre este texto.

Una vez el usuario haya iniciado sesión, se le cargará la vista principal, la cual se puede decir que es un menú con las distintas opciones.

La primera opción que aparece es el historial de riegos, en la cual se puede ver un registro de los riegos anteriores, en los que se especifica la hora del riego, la fecha y otros valores como humedad y cantidad total de agua.

La siguiente opción, corresponde al estado del riego, en esta vista se puede comprobar si el riego está programado o si lo acabamos de accionar por si queremos regar en un momento determinado.

Después viene la opción de la configuración del riego, que es complementaria a la anterior, ya que en ella podemos ver los valores que van marcar si se va a regar o no. En esta vista tenemos un enlace a la vista “editar sector”, la cual no está definida en el menú, solo es accesible desde la configuración del riego. En editar el sector, podemos cambiarle el nombre y la configuración del riego.

La siguiente opción que tenemos es la de solicitar nuevo horario, en esta vista el usuario puede enviar al administrador una nueva solicitud de riego, una vez la solicitud haya sido enviada, al administrador le llega un email notificándole que el usuario quiere dar de alta una nueva configuración de riego. Una vez el administrador haya aceptado la propuesta, el usuario podrá verla en el listado de configuraciones de riego.

Por último, está la opción de reportar incidencia, en esta vista, el usuario puede enviar un correo electrónico al administrador comentándole alguna incidencia que haya tenido y que se tenga que arreglar.

2.3. Estilo de la web

El estilo de la web, está basado en unos colores “naturales”, es decir, adaptados a la temática del proyecto como es el riego.

La fuente que se ha usado para toda la web es Roboto desarrolla por Google y perteneciente al tipo “sans-serif”.

Para la web, sean usado cinco colores:

El color verde que está presente en toda la web es #4FBF6A, este color se ha usado, tanto para texto, como para relleno de los botones.

El color rojo, cuyo código es #FF0000, se ha usado para los botones de rechazar/cancelar y para “cerrar sesión”.

El color grisáceo, que está presente tanto en la barra de navegación como en las tablas, corresponde al siguiente código #353535.

El color azul, solo es presente en el menú de navegación para identificar dónde se va a pulsar, su código es #00B9FF.

El color blanco, que es el que existe por defecto en una web.

3. CONFIGURACIÓN DEL SERVIDOR

El servidor ha sido montado en una Raspberry Pi 4, se le ha instalado el sistema operativo Raspbian en una microSD de 128 gigabytes de capacidad.

Al servidor se le ha instalado un paquete “LAMP”, es decir Apache, PHP en su versión 7.4, MariaDB 5.0.6 y phpMyAdmin.

Se ha instalado la versión 7.4 de PHP para así poder tener Symfony en la versión 5.0.

Una vez instalado el paquete LAMP, se ha instalado Symfony y GIT siguiendo los métodos de la documentación (indicada en la bibliografía).

La configuración de Apache, se ha hecho de acuerdo al protocolo https, es decir se ha redirigido todo el tráfico a una conexión cifrada. El certificado ha sido autofirmado y autogenerado.

Respecto a la visualización de la web fuera de la red local, se ha usado un dominio el cual se ha redireccionado a la IP Pública de la red. Una vez hecho esto, en la configuración del router se ha desviado todo el tráfico entrante por http y https hacía la dirección del servidor.

3.1. Estructura de la base de datos

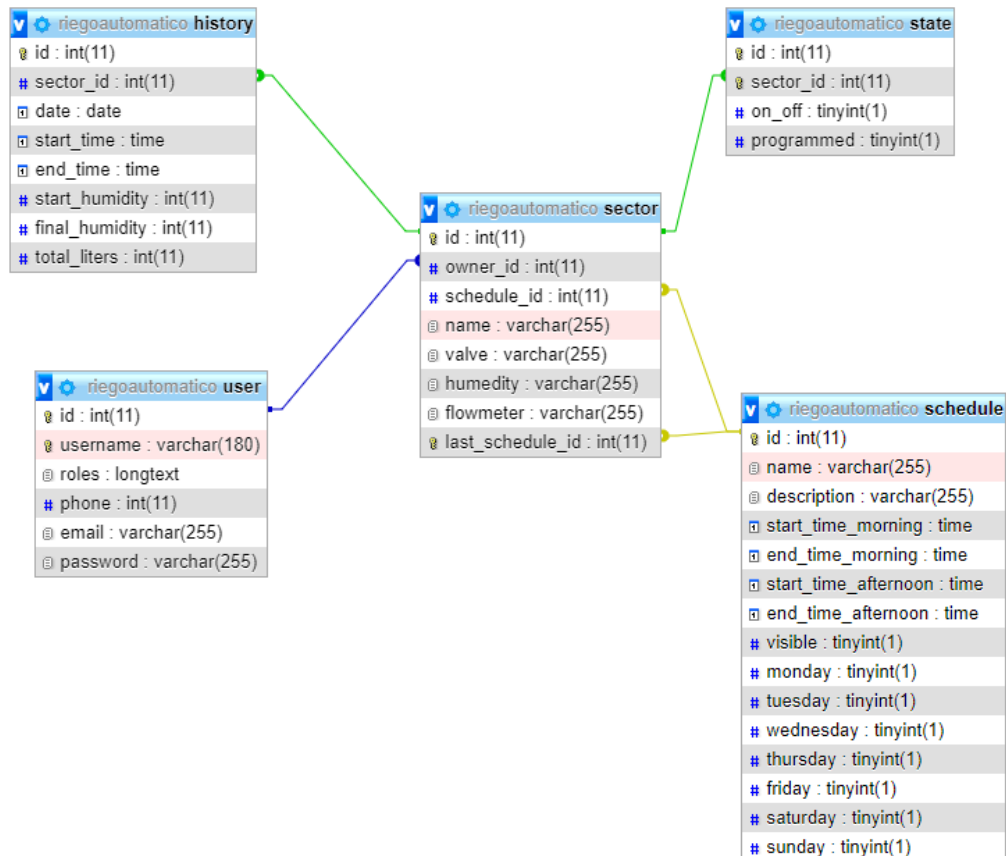
En la base de datos únicamente hay una tabla por cada entidad, sin tablas intermedias, ya que las relaciones que existen son Uno a Uno y Uno a Muchos.

Un usuario tiene muchos sectores.

Un sector tiene muchos historiales.

Un sector tiene un estado.

Un sector tiene un horario.



4. CONFIGURACIÓN DEL ARDUINO

Por último, el tercer componente del proyecto es un microcontrolador Arduino, el cual es el encargado de activar y desactivar el riego a parte de tomar los datos más relevantes de este.

El microcontrolador que se ha usado ha sido un Arduino Mega 2560, ya que tiene más memoria y más entradas/salidas respecto a una placa básica como puede ser Arduino Uno. De esta manera, usando dicha placa si se quisiera aumentar el número de sectores no habría problema, ya que hay memoria suficiente.

Para poder conectar el microcontrolador a la base de datos, se ha acoplado una placa de expansión de ethernet al Mega 2560.

La programación del microcontrolador se ha basado en consultas a la base de datos mediante la librería `My_SQL_Connector_Arduino` de ChuckBell.

A grandes rasgos, la estructura del bucle del Arduino es la siguiente:

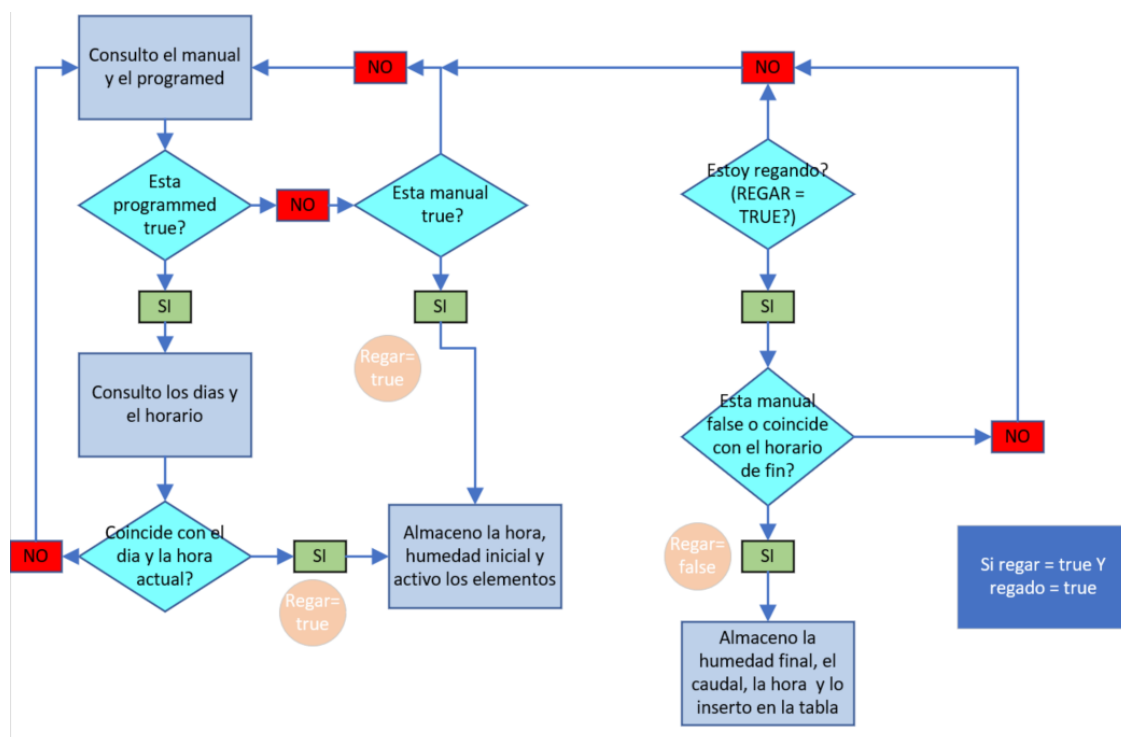
- Consultar el estado del sector

- Si está en modo programado consultar el horario, si coincide activar riego.

- Si está en modo manual activar riego.

- Si está activo el riego, activar válvulas y leer estado de humedad y caudal.

- Si estaba el riego activo y se ha desactivado, apagar todo y guardar datos.



4.1. Componentes usados

Para el sistema de riego, se han usado los siguientes componentes:

Placa solar

Regulador de carga solar

Batería 12v

Arduino Mega 2560

Ethernet Shield

Electroválvula 12V

RTC DS3231

Relé 12V

Relé 5V

Caudalímetro 5V YF-S201

Higrómetro YL-69

5. BIBLIOGRAFÍA

Todas las páginas que han sido consultadas durante la realización del proyecto son las siguientes:

Para la configuración de Symfony en el servidor:

https://symfony.com/legacy/doc/jobee/1_2/es/01?orm=Doctrine

<https://symfony.com/doc/4.2/reference/requirements.html>

<https://www.osradar.com/how-to-install-composer-on-linux/>

Para la conexión entre la base de datos y el microcontrolador:

https://github.com/ChuckBell/MySQL_Connector_Arduino

Para la configuración del caudalímetro en el microcontrolador:

<https://www.luisllamas.es/caudal-consumo-de-agua-con-arduino-y-caudalimetro/>