# Capítulo 10 – Desarrollo del SAR

El desarrollo del SAR se descompone en varios niveles de capas. A su vez, existen dos esquemas muy diferenciados el lógico y el físico (Ilustración 1 - Esquema general del SAR).

El esquema físico se compone de los dispositivos electrónicos que controlan los actuadores, efectores y sensores. Además de los microcontroladores (Arduino Mega, Arduino Nano) y la microcomputadora Raspberry Pi 3.

El esquema lógico se compone del sistema operativo Raspbian y una aplicación web desarrollada bajo la arquitectura cliente/servidor respetando el conjunto de herramientas MEAN. Además, contiene un administrador de servicios para Node denominado PM2, y un controlador de cámaras de video conectadas al Sistema Operativo llamado Motion.

Este esquema a su vez se encuentra organizado en dos unidades funcionales llamadas Back-end y Front-end.

Ilustración 1 - Esquema general del SAR

## **10.1 Desarrollo de la aplicación (front-end)**

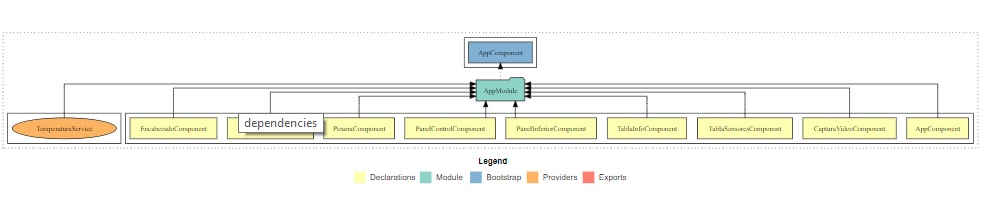
El front-end se encuentra desarrollado en Angular 4+, contando con los siguientes esquemas (Ilustración 2 - Módulos Angular):

Ilustración 2 - Módulos Angular

Al conectarnos al servidor, desde un cliente (navegador web) nos descarga la aplicación embebida, y la comunicación con el servidor sucede a través de Json.

## **10.2 Desarrollo del servidor (back-end)**

El back-end, construido en Node y Express, almacena los estáticos del front-end y los suministra al conectarse un cliente. Los directorios del proyecto los podemos observar en la ilustración (Ilustración 2 - Backend)

Ilustración 3 - Backend

En el directorio Routes, podemos apreciar dos archivos: Api.js y Placas.js. El primero se encarga de gestionar las llamadas por métodos HTTP, implementando REST. Por otro lado, Placas.js se encarga de gestionar la conexión a los periféricos de la Raspberry, que son la Arduino NANO y la Arduino MEGA.

Además, en el directorio dist, se almacena el compilado (js, html, css3) de la aplicación angular y en la raíz del directorio, existe el archivo server.js encargado del despliegue del servidor en el puerto 3000, utilizando node.

La conectividad a MongoDB es realizada a través del driver MongoClient definida en el archivo api.js.

A su vez, Express filtra las rutas accedidas al host (192.168.2.1:3000) resolviendo y respondiendo las distintas consultas realizadas por la App en Angular.

Las distintas llamadas resueltas por Express, utilizando REST son:

* Temperturas 🡪 Api.js, devolviendo temperaturas almacenadas en la BD.
* Monoxidos 🡪 Api.js, devolviendo monóxidos almacenadas en la BD.
* Apagar 🡪 Api.js, apagando la Raspberry desde el sistema operativo.
* Reiniciar 🡪 Api.js, reiniciando la Raspberry desde el sistema operativo.
* Arriba, Abajo, Izquierda, Derecha, Stop 🡪 placas.js, ejecutando un orden a los motores para desplazarse o detenerse.
* Ultrasonido 🡪 placas.js, obteniendo los valores de ultrasonido detectados.
* GPS 🡪 placas.js, obteniendo los valores como altura, velocidad, curso, fecha y coordenadas.
* Monoxido 🡪 placas.js, se obtiene el valor actual de monóxido en el ambiente.

Ver anexo de códigos (Api.js, Placas.js, Server.js)

## **10.3 Esquema de la arquitectura lógica**

La combinación del back-end (servidor) con el front-end (cliente) permiten el funcionamiento de la aplicación web. En el siguiente gráfico (Ilustración 4 - Arquitectura lógica del SAR) se puede apreciar la arquitectura lógica cliente/servidor que posee el SAR, detallando todos los componentes, tanto de software como de hardware, que intervienen para que el sistema funcione.

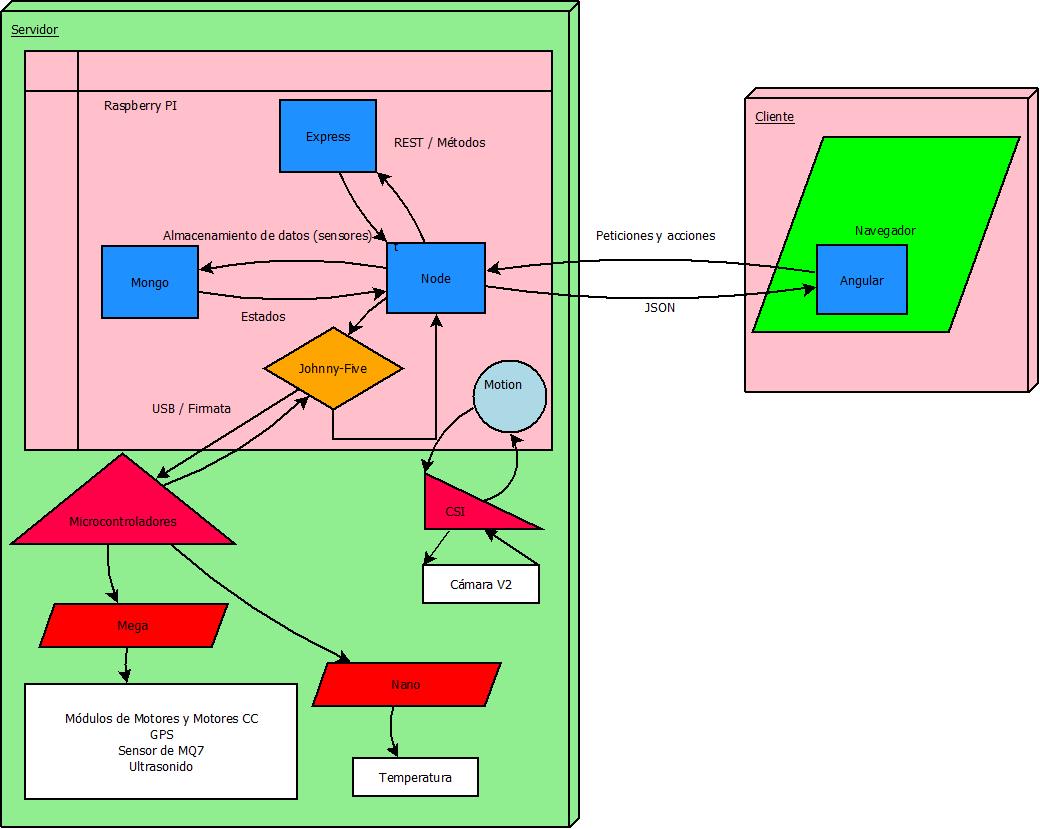


Ilustración 4 - Arquitectura lógica del SAR

## **10.4 Funcionamiento de la App**

Para poder acceder a la aplicación del SAR y controlar el robot móvil, se debe conectar al punto de acceso (AP, access point) de red WIFI que genera la Raspberry denominada “SAR”, con su respectiva contraseña.

Una vez conectado a la red, se debe acceder mediante un navegador web a la URL:

**192.168.2.1:3000**

Puerto

IP

Automáticamente se desplegará la interfaz que se puede apreciar en la siguiente ilustración (Ilustración 6 - Aplicación web).

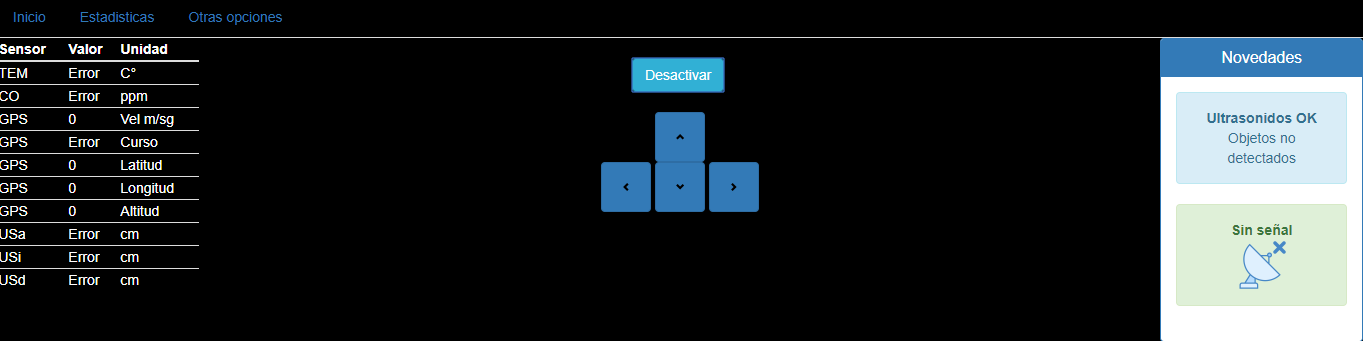


Ilustración 5 - Aplicación web

Como se puede ver, la interfaz cuenta con un menú, con tres opciones.

Estas opciones son: Inicio, estadísticas y otras opciones.

La solapa de inicio, se encuentra dividida en tres partes. En el panel izquierdo, se observa una tabla de valores obtenidos por los sensores. Estos valores se actualizan en tiempo real. En el panel central se encuentra, un botón para activar/desactivar la visualización de video en tiempo real. Por debajo, de dicho botón, existe un conjunto de botones, que permiten controlar el movimiento y la dirección del robot móvil. En el panel derecho, veremos una sección de “novedades” la cual muestra los distintos estados de los sensores con mensajes de advertencia y/o información al usuario.

En la solapa estadísticas, se visualizan dos diagramas estadísticos. El primero es de temperatura (referencia falta captura), que a través de unos selectores podemos indicar la fecha de inicio y de fin. De esta forma se confecciona una gráfica, al cliquear en obtener, siempre y cuando existan datos almacenados en el período seleccionado. La gráfica muestra en las columnas, los días y en las filas, las horas del día desde las 00:00 hs hasta las 23:00. En cada una de las intersecciones fecha/hora se muestra el promedio de temperaturas en un rango de 1 hr. En la primera columna, se genera una media de los valores que se encuentran en la misma fila adyacente de las fechas seleccionadas.

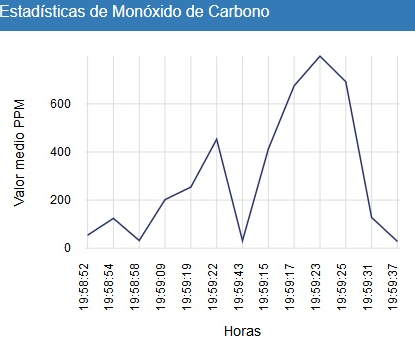


Ilustración 6 - Estadísticas de monóxido

En el segundo (Ilustración 7 - Estadísticas de monóxido), se toman valores en tiempo real del valor de monóxido de carbono presente en el ambiente. Los mismos se representan según el horario y el valor en PPM obtenido. Por otro lado, se permite colocar el máximo de valores a visualizar.

En la solapa otras opciones (Ilustración 8 - Aplicación web - Otras opciones), visualizamos dos botones, que permiten apagar o reiniciar el sistema operativo, por ende, del robot móvil.

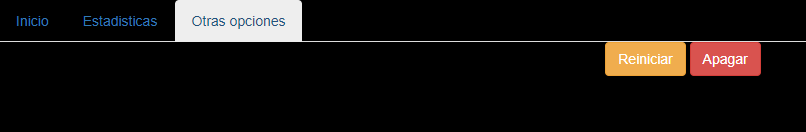


Ilustración 7 - Aplicación web - Otras opciones

## **10.5 Puesta en producción del SAR**

Para desplegar el SAR se necesitó de software y configuraciones adicionales que a continuación se describen.

### 10.5.1 Configuración de Raspberry como AP

El SBC Raspberry Pi 3 se encuentra dotado de un módulo wifi que permite la configuración como modo Access Point (AP). Para que diversos dispositivos se puedan conectar al SAR, se realizó la configuración de dicho módulo de la siguiente manera. [2]

Primeramente, se instalaron, mediante Raspbian, dos programas necesarios para generar el punto de acceso (AP), **hostapd** y **isc-dhcp-server**. Introduciendo, en una terminal de Raspbian, los siguientes comandos:

sudo apt-get install hostapd isc-dhcp-server

A continuación, se configuro el archivo que permite la configuración de DHCP (Glosario), accediendo al mismo de la siguiente manera:

sudo gedit /etc/dhcp/dhcpd.conf

Dentro del mismo se comentaron dos líneas que permiten definir el nombre de dominio deseado, dado que no se utilizaron:

#option domain-name "example.org";

#option domain-name-servers ns1.example.org, ns2.example.org;

Se descomento la línea que a continuación se resalta, que permite definir a la Raspberry como servidor DHCP:

# If this DHCP server is the official DHCP server for the local

# network, the authoritative directive should be uncommented.

authoritative;

Finalmente, la última configuración realizada con este archivo fue la de definición de la LAN a generar, donde se configuraron los siguientes parámetros:

subnet 192.168.2.0 netmask 255.255.255.0 {

range 192.168.2.10 192.168.2.30;

option broadcast-address 192.168.2.255;

option routers 192.168.2.1;

default-lease-time 600;

max-lease-time 7200;

option domain-name "local";

option domain-name-servers 8.8.8.8, 8.8.4.4;

}

El siguiente archivo a modificar fue el isc-dhcp-server, para ello se accedió al mismo de la siguiente manera:

sudo gedit /etc/default/isc-dhcp-server

En él se configuró la interfaz de las Raspberry que funcionara como servidor DHCP, en nuestro caso la wlan0, agregando la misma en la siguiente línea:

INTERFACES=”wlan0”

Para establecer el acceso a la aplicación web del SAR, se precisó establecer una ip (Glosario) estática a la interfaz wlan0 de la Raspberry, para ello se realizó lo siguiente.

Primero se debió desactivar la interfaz wlan0, de la siguiente manera:

sudo ifdown wlan0

Luego se modificó la interfaz accediendo al siguiente archivo:

sudo gedit /etc/network/interfaces

En él se agregó la ip estática definida para el SAR y se comentaron las tres últimas líneas, que definían la configuración manual de la interfaz:

allow-hotplug wlan0

iface wlan0 inet static

address 192.168.2.1

netmask 255.255.255.0

#iface wlan0 inet manual

#wpa-roam /etc/wpa\_supplicant/wpa\_supplicant.conf

#iface default inet dhcp

Luego de guardado el archive anterior se asignó y habilitó las ip mencionada con el siguiente comando en la terminal:

sudo ifconfig wlan0 192.168.2.1

Para finalizar la configuración AP, se procedió a la modificación del siguiente archivo:

sudo gedit /etc/hostapd/hostapd.conf

El cual permite la definición de los ajustes básicos de la red wifi a crear, como el SSID, contraseña de acceso, interfaz y demás opciones que a continuación se detallan:

interface=wlan0

ssid=sar

hw\_mode=g

channel=6

macaddr\_acl=0

auth\_algs=1

ignore\_broadcast\_ssid=0

wpa=2

wpa\_passphrase=sartesis2017

wpa\_key\_mgmt=WPA-PSK

wpa\_pairwise=TKIP

rsn\_pairwise=CCMP

Para que se inicie el modo AP al arrancar el sistema operativo se crearon los siguientes Daemon (Glosario):

sudo service hostapd start

sudo service isc-dhcp-server start

Luego, se habilitaron dichos Daemon con los siguientes comandos:

sudo update-rc.d hostapd enable

sudo update-rc.d isc-dhcp-server enable

### 10.5.2 Configuración del servicio Motion

## Como ya se mencionó anteriormente (capitulo 8.3 Selección tecnologías software), el software seleccionado para la captura de imágenes por parte de la cámara de Raspberry es Motion. Para poder utilizar dicho programa se tuvo que realizar los siguiente.

Primero, se habilitó el uso de la cámara de Raspberry. Para ello, se accedió a las herramientas de configuración de Raspberry con el siguiente comando en una terminal de Raspbian:

sudo raspi-config

Dentro de la lista de opciones se activó la cámara seleccionando la 5:

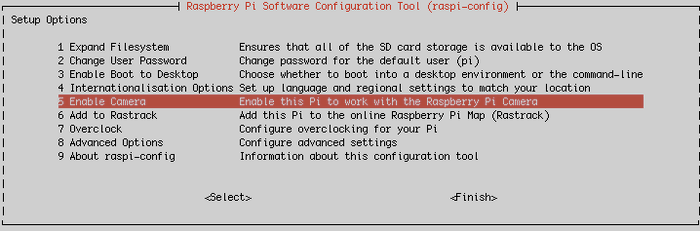


Ilustración 8 - Software de configuración de Raspberry

Seguidamente se instaló Motion de la siguiente manera:

sudo apt-get install -y motion

Este software automáticamente detecta la cámara de la Raspberry y captura las imágenes obtenidas por la misma. A su vez genera un servicio para poder visualizar en tiempo real la captura de video en el ip predeterminado de la SBC y un puerto por defecto.

Para configurar dichos parámetros, Motion cuenta con un archivo que permite realizar los distintos ajustes del software. Para acceder al mismo se ejecuto el siguiente comando:

sudo gedit /etc/motion/motion.conf

En el cuál se modificaron los siguientes parámetros:

videodevice /dev/video0

width 256

height 192

threshold 7000

minimum\_motion\_frames 2

quality 75

webcam\_port 9081

DAEMON = ON

Webcam\_localhost = off

Luego, para aplicar los cambios se debe reiniciar Motion de la siguiente manera:

sudo service motion restart

Con lo que para poder visualizar la captura de la cámara se debe acceder, mediante un navegador web, a la combinación ip:puerto configurada previamente para la Raspberry:

**192.168.2.1:9081**

Finalmente se modificó el siguiente archivo:

sudo gedit /etc/default/motion

Para que el servicio de Motion arranque al iniciar Raspbian. En el mismo se modificó la siguiente línea:

start\_motion\_daemon = yes

Con esta línea, puesta como “yes” (por defecto es “no”), habilitamos el Daemon de Motion.

### 10.5.3 Instalación del gestor de procesos PM2

Node es administrado por PM2 (administrador de procesos para JavaScript) el cual inicia el servidor, automáticamente al arrancar Raspbian, controlando y monitoreándolo. En la siguiente captura (Ilustración 9 - Monitor de PM2) podemos apreciar como el gestor de procesos PM2, permite monitorear, controlar y ejecutar la aplicación definida (server.js) que realiza el despliegue del servidor del SAR. En caso de fallos, PM2 re arranca la aplicación emitiendo mensajes. Además, permite monitorear cantidad de reinicios, generar un log, mostrar el tiempo de carga de la aplicación, como reiniciarla, pararla y listar todos los procesos que gestiona.

Por otro lado, permite generar el archivo de startUp(COLOCAR REFERENCIA) para iniciar con el sistema operativo de Raspbian. Este archivo y los pasos de generación se incluirán en el anexo de códigos.

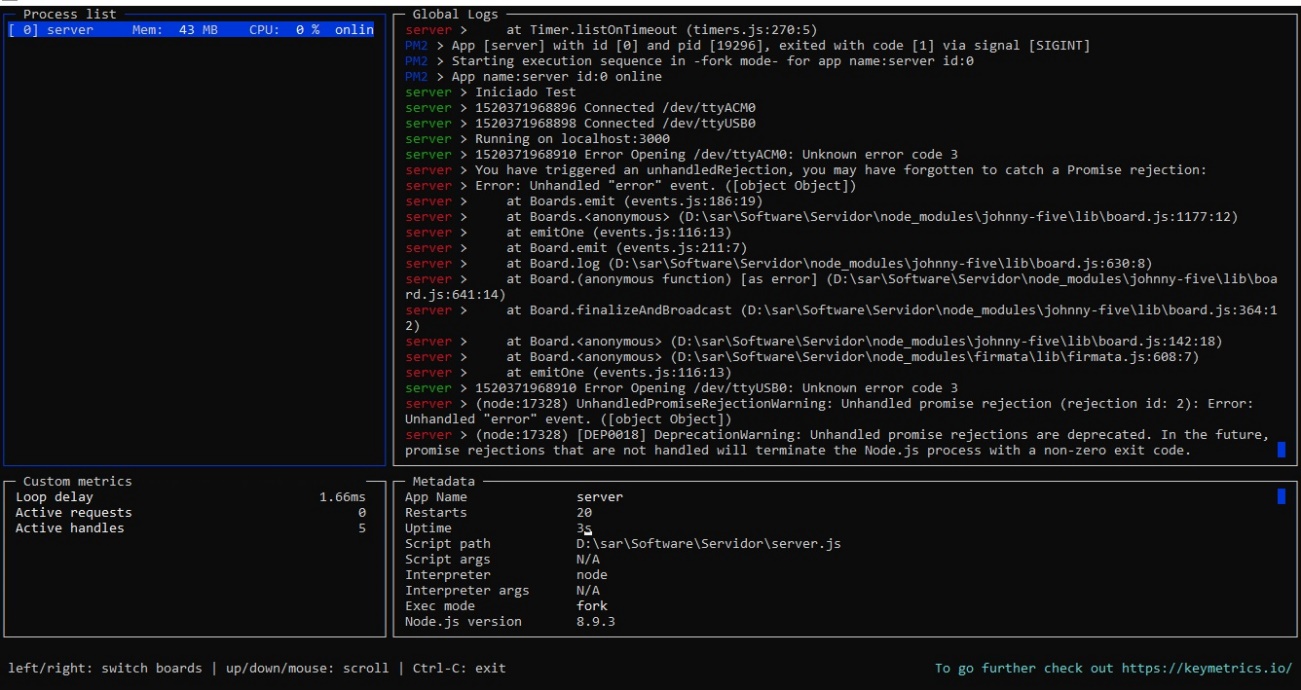


Ilustración 9 - Monitor de PM2

Otra característica de PM2, es la integración con Keymetrics (Ilustración 10 - Keymetrics), donde si el servidor tiene acceso a Internet, es posible hacer el seguimiento desde internet con un token de seguridad.



Ilustración 10 - Keymetrics

## **Resumen**

Este capítulo explicó la utilidad tanto del back-end como del front-end, necesarios para el funcionamiento de la aplicación web que despliega el SAR.

A su vez se describieron los pasos realizados para configurar las Raspberry Pi como un punto de acceso (AP) inalámbrico, lo cual permite que diversos dispositivos se puedan conectar al robot móvil, mediante un ip y puerto previamente configurados, para poder manipularlo.

Por otro lado, se describió la configuración del software utilizado para la captura de imagen y video, de la cámara del SAR, denominado Motion.

Finalmente se describe la utilidad de un programa de gestión de procesos (PM2), que sirve como administrador de Node y permite monitorear el funcionamiento de la aplicación web.