

PIÑASO S.L.

ASIR

SALESIANOS ATOCHA



PIÑASO

2019

David Cerdón Anguita

Sergio Pérez Sandeogracias



Índice

1. INTRODUCCIÓN.....	3
1.1. Objetivo	3
1.2. Justificación.....	3
1.3. Análisis de lo Existente	3
1.4. Propuesta Detallada	4
2. PLANIFICACIÓN.....	4
3. DISEÑO.....	5
3.1. Empresa	5
3.1.1. IDENTIDAD CORPORATIVA	5
3.1.2. RESPONSABILIDAD SOCIAL	6
3.1.3. FORMA JURÍDICA (SOCIEDAD LIMITADA)	6
3.2. Aplicación	7
3.2.1. DISEÑO ARQUITECTÓNICO.....	11
3.3. Dispositivo	12
3.3.1. DISEÑO ARQUITECTÓNICO.....	16
3.4. Web.....	17
3.4.1. DISEÑO ARQUITECTÓNICO.....	19
4. CONCLUSIÓN.....	20
5. BIBLIOGRAFÍA.....	21
5.1. Módulo MPU-9250.....	21
5.2. Bluetooth.....	21
5.3. Kodular.....	21



1. Introducción.

1.1. Objetivo

Desde Piñaso S.L. somos conscientes del gran problema que existe en nuestras carreteras. Los moteros somos aquellos que estamos más desprotegidos. El gran problema es que no existe una “chapa” que nos pueda proteger al caernos. Lo más importante para salvar la vida, o incluso miembros, es la rapidez de los servicios sanitarios.

Por eso, desde Piñaso S.L., queremos que esto deje de ser un gran problema. Nuestra proposición es de un dispositivo que sea capaz de detectar cuando nos hemos caído y avisar a uno de nuestros contactos de nuestra situación y posición exacta, para así poder actuar de manera más inmediata, salvando vidas.

1.2. Justificación

Nuestro objetivo a la hora de plantear el proyecto era que teníamos que demostrar todo lo aprendido durante el módulo: **HTML, CSS, JavaScript, PHP, MySQL, Apache**, Administración de S.O. **Raspbian y Hardware**. Pero, a su vez, queríamos ver nuestras capacidades de autodidactas. Con esto nos pusimos manos a la obra y llegamos a adquirir los nuevos conocimientos, entre los que se encuentran: **Python, Comunicación** vía **Bluetooth**, Crear **Dispositivos Virtuales**, **JQuery, Bootstrap**, Conocimientos físicos que afectan a los sensores **Acelerómetro, Magnómetro y Giroscopio, Comunicación Serial, FireBase** y **Programación** en **Kodular** (Java)

1.3. Análisis de lo Existente

Actualmente, en el mercado no existe nuestro producto como tal, es cierto, que existen diferentes dispositivos, pero están mas orientados a cuando la moto es robada, o esta se cae, cosa que creemos que esto es un error, por que al caernos la moto puede ir en una dirección muy distinta a la de nuestro cuerpo. Aparte, estos tienen un precio mucho más elevado, rondando los 200€ de inversión inicial y una suscripción mensual, mientras que Piñaso sería un coste total de 80€. Por tanto, nuestra competencia sería: SmartBike y OnBike.



1.4. Propuesta Detallada

La idea sería ofrecer una utilidad de por vida, con intenciones de ser un dispositivo que termine por ser indispensable para todo tipo de motero o conductor de moto.

En la compra vendría un dispositivo. Piñaso, y un código QR con el cual se podrá descargar la aplicación.

El dispositivo es una Raspberry Pi Zero W con el sensor MPU9250 (módulo que incorpora un giroscopio, magnetómetro y acelerómetro). En la Raspberry hay un código Python que estaría todo el rato comprobando el estado del módulo MPU, en cuanto se cumplan los requisitos implementados entenderá que tiene que enviar la señal a la APP, esto lo realiza mediante un dispositivo virtual creado en la Raspberry, lugar donde nos conectamos vía Bluetooth. Una vez ya en la APP, comprobará la señal recibida, si en esa señal recibe un “pinaso” significa que la orden es correcta. En ese caso, la aplicación escribirá un mensaje SMS al contacto que se haya registrado previamente, con nuestra dirección y nuestra posición de Latitud y Longitud.

También, se ha realizado una página web donde se podrá comprar el Dispositivo, existen dos maneras de comprarlo, ya sea el dispositivo terminado, montado e instalado (KIT) o comprar los componentes por piezas (DIY) y que sea el usuario el que lo haga por su cuenta, en ambos casos la aplicación siempre será la versión definitiva.

2. Planificación

Dado que esto es un proyecto de ASIR, la idea era que lo tuvieses todo operativo con las funcionalidades mínimas antes de la segunda quincena de marzo. Teniendo así listo el dispositivo, la App y la página Web.

Dado que este proyecto tiene muchas mejoras, implementaciones nuevas, ampliaciones... se tendría que tener en constante actualización y mantenimiento, por tanto, esto es un proyecto con grandes visiones de futuro.



3. Diseño

3.1. Empresa

3.1.1. Identidad Corporativa

Distinguiremos tres elementos:

- Misión:

Nuestra misión es cuidar de las personas en las carreteras del mundo, atendiendo su salud y bienestar, apoyándonos en una estructura de excelente calidad y con los mejores medios tecnológicos para el desarrollo de nuestro producto.

Por ello, incentivamos la máxima profesionalidad en nuestra actividad diaria con nuestro personal comprometido y vocacional, así como la innovación y la actividad investigadora y docente.

- Visión:

Convertirnos en una empresa referente en el sector de la seguridad vial en España, y por ello poder llegar a ser reconocidos a nivel mundial y así fomentar la investigación en la seguridad vial, su atención integral y su innovación.

- Valores:

La ética, como principio rector de nuestras decisiones.

Compromiso y concepción integral de lo que desarrollamos, para poder satisfacer de forma eficiente las necesidades asistenciales de nuestros clientes, los motoristas.

Calidez, con el fin de poder ofrecer un trato humano respetuoso, digno y confidencial.



3.1.2. Responsabilidad Social

- Responsabilidad con los trabajadores:

Respetar los derechos humanos de nuestros empleados, impulsar una cultura de respeto de los derechos humanos entre los empleados, promoviendo para ellos formaciones específicas e implementando herramientas para su ayuda.

Nuestra normativa interna está basada en el respeto con los Principios Rectores sobre las Empresas y Derechos Humanos de las Naciones Unidas.

- Responsabilidad con el medio ambiente:

Las actividades de Piñaso SL se realizarán de la manera más respetuosa con el medio ambiente, favoreciendo la conservación de la biodiversidad en las áreas de influencia de nuestros proyectos.

- Responsabilidad con la sociedad:

En Piñaso SL trabajamos de la mano con instituciones y socios que son referentes nacionales a nivel de seguridad vial, como por ejemplo Ponle Freno, Fundación Mapfre, RACE, entre otros.

- Responsabilidad con el cliente:

Respetar los derechos humanos de nuestros clientes. Respetar en todo momento el derecho a su intimidad, protegiendo y haciendo un uso correcto de los datos confiados por nuestros clientes.

3.1.3. Forma Jurídica (Sociedad Limitada)

- Ventajas

El capital Mínimo para constituir la empresa son 3000€. Una vez constituida la sociedad, puedes invertir el dinero en su puesta en marcha.

La delimitación de responsabilidades entre los socios es fácil de organizar.

La incorporación de nuevos socios y transmisión de acciones a terceros.

- Desventajas

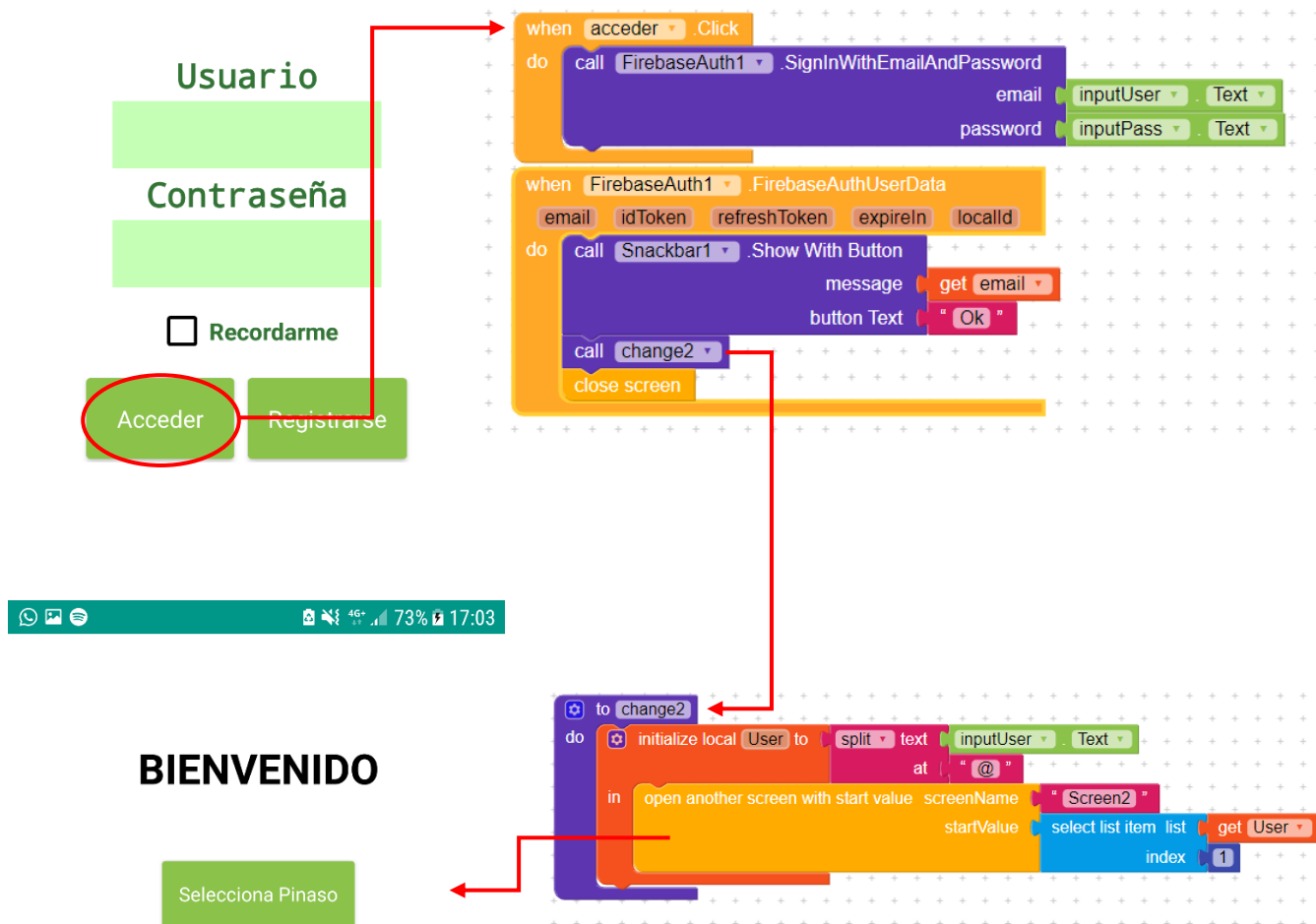
El reparto equitativo de acciones podría hacer inviable la continuidad y el funcionamiento de la empresa.









3.2. Aplicación

Para la realización de la Aplicación se ha utilizado Kodular. Kodular es una herramienta basada en App Inventor, es una plataforma que permite la programación de Apps de manera rápida y ágil, guardando el código en su propia nube para poder acceder en cualquier momento a tu app. La programación se realiza mediante módulos y complementos.

Al iniciar la aplicación aparece una imagen con el Logo de la compañía, una vez clicado aparece una pantalla del Log-In.



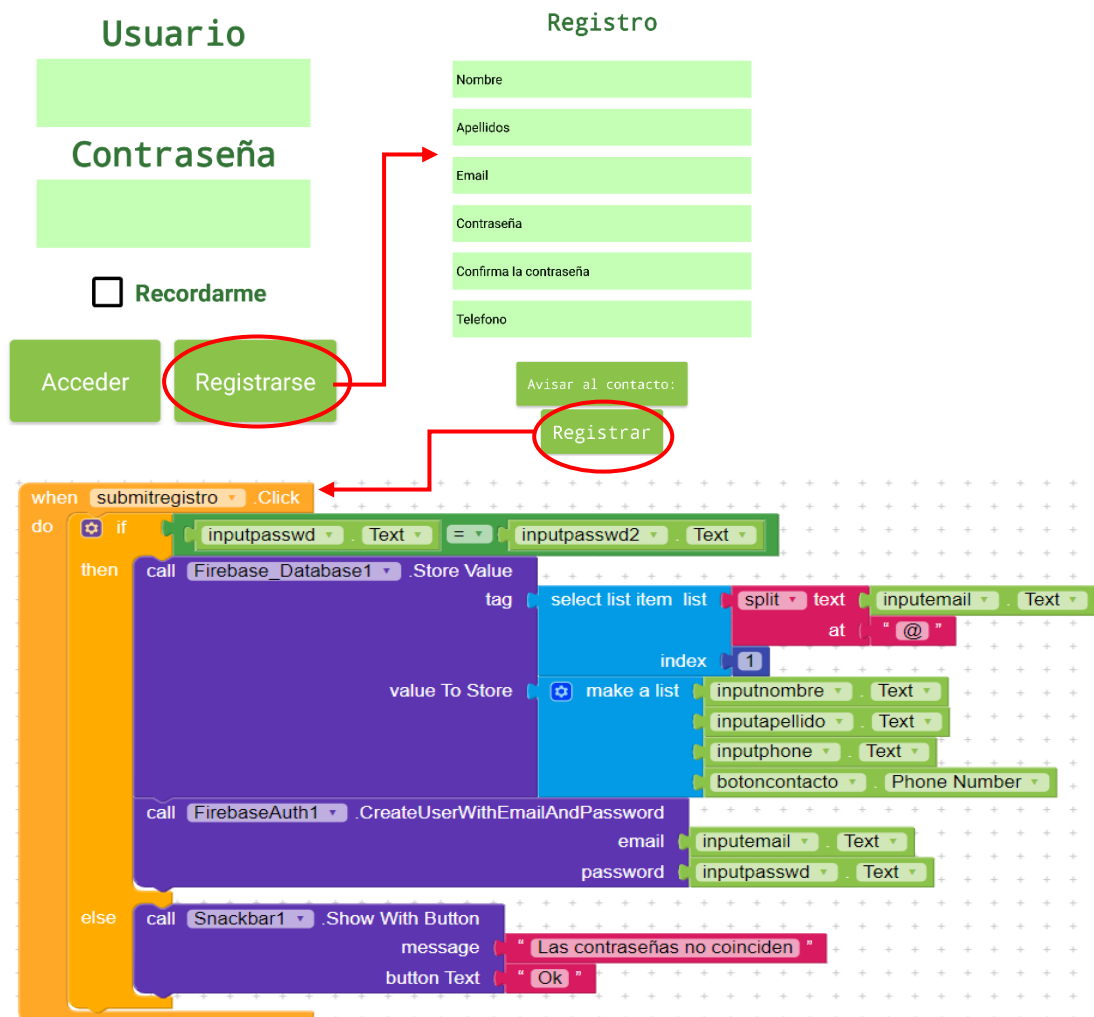


<div>  Buscar por dirección de correo electrónico, número de teléfono o UID de usuario </div> <div> Añadir usuario   </div>				
Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
mola@gmail.com		11 mar. 2019	11 mar. 2019	SJ2gXLyJtfRgtZb52cMBzJsJPYE2
keko@gmail.com		6 mar. 2019	6 mar. 2019	SaKeDQK4lQgoRat5rTCfP8y3yTH3
pinaso@gmail.com		5 mar. 2019	11 mar. 2019	xppdcZ1cATXVMGmiha25fCvRtYh2

Filas por página: 50
 1-3 de 3

Como se puede observar, se puede introducir tu Correo y contraseña para acceder a las funcionalidades de la App, la aplicación realizará una llamada a Firebase para comprobar si en nuestra base de datos está ese usuario ya registrado, en caso afirmativo se cerrará la pantalla en la que nos encontramos y abrirá una nueva donde mantendrá los datos de inicio de sesión.

También tenemos la opción de poder registrarnos, donde se tendrá que introducir tu nombre, apellidos, email, contraseña, tu teléfono y elegir al contacto que se quiera avisar en caso de accidente.





Una vez registrados, tendríamos que seleccionar que dispositivo es nuestro Piñaso.

BIENVENIDO

Selecciona Pinaso

Uconnect
1C4BU0000FPC01024

smart Media-System

Mifa_10

MY CAR

B8:27:EB:8E:64:C3 raspberrypi

MI Band 2

```

when ListaButton Click
do
  set ListaDev Visible to true
  set ListaDev Elements to ClienteBluetooth1 Addresses And Names
  set ElegirPinaso Visible to false

when ListaDev After Picking
do
  set global User to trim select list item list index 1
  set global selection to ListaDev Selection
  set ListaDev Visible to false
  set ElegirPinaso Visible to false
  set Mapa Visible to true
  if call ClienteBluetooth1 Connect address get global selection
  then
    set Clock1 Timer Enabled to true
    set bluetooth Timer Enabled to true
  else
    set Welcome Text to "error"
    set Clock1 Timer Enabled to false
    set bluetooth Timer Enabled to false
  
```

Cuando lo hemos seleccionado y se termina de enlazar aparecerá la siguiente pantalla. Donde al darle al botón, nos llevara, en el mapa a la posición exacta de donde nos encontramos.

Donde Estoy

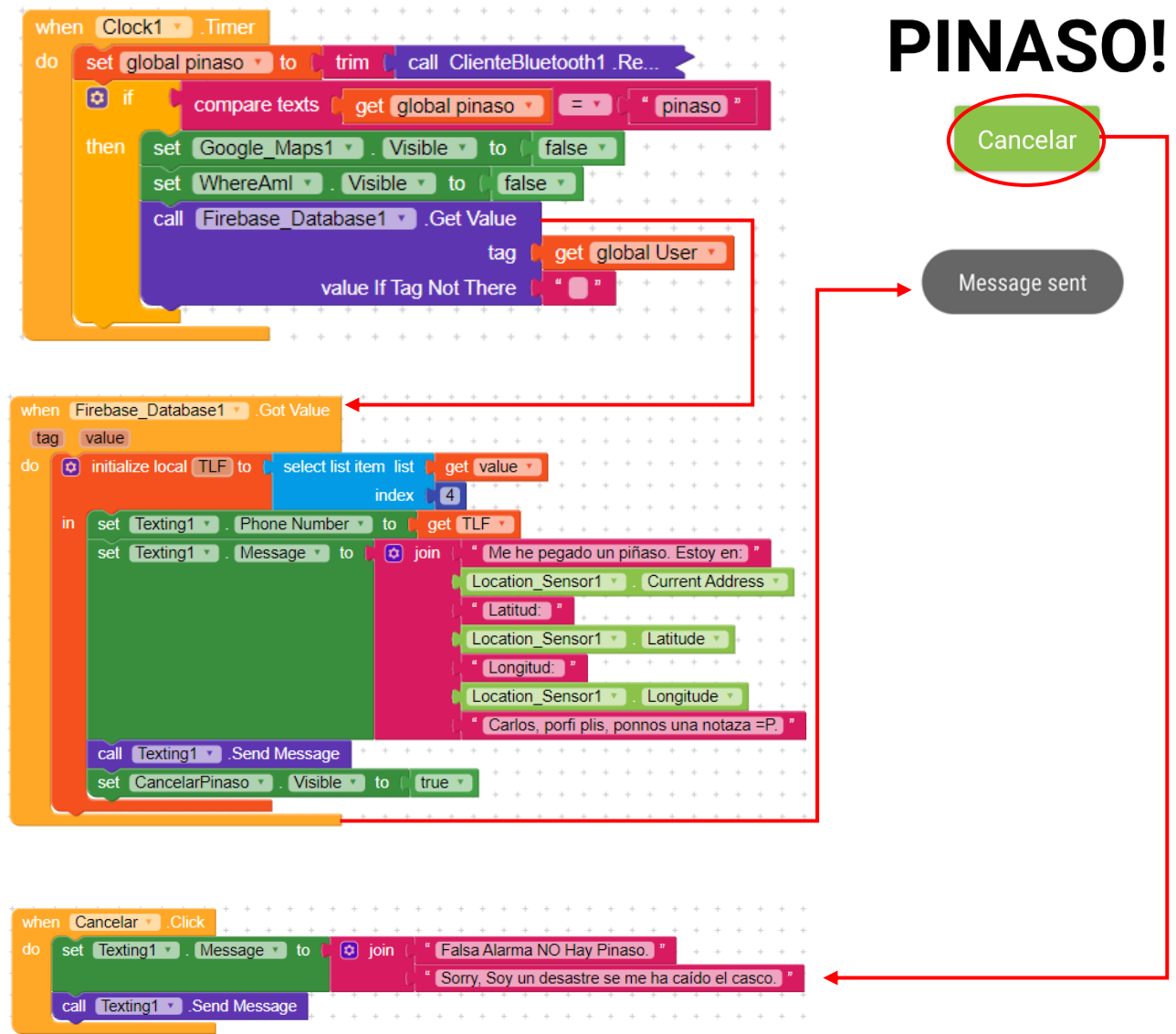
```

when WhereAml Click
do
  call Google_Maps1 Move Camera
  lat Location_Sensor1 Latitude
  lng Location_Sensor1 Longitude
  zoom 10
  
```

Donde Estoy

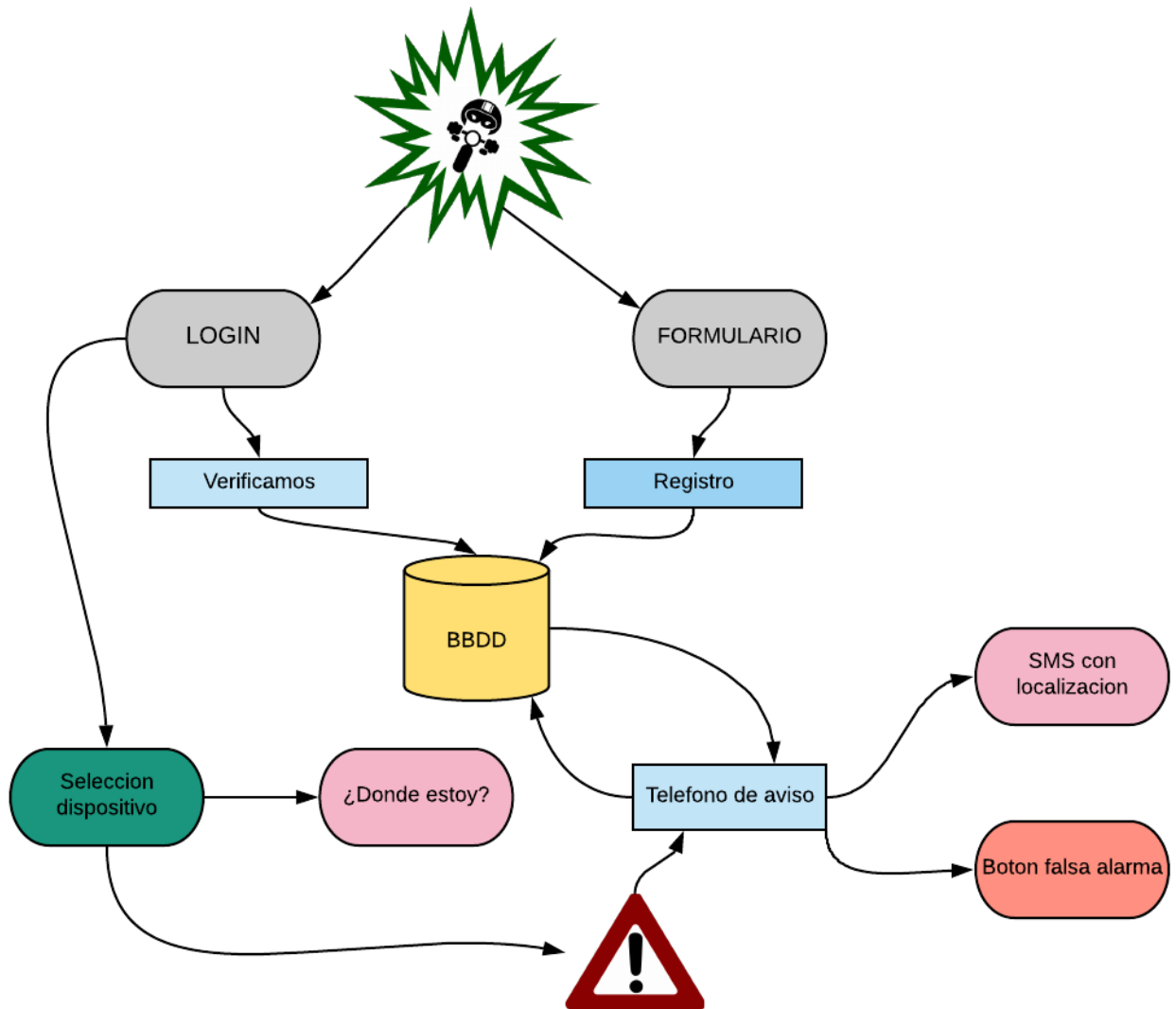


En cuanto el dispositivo detecte el impacto, enviará una señal a nuestra aplicación. Al recibir “pinaso” la aplicación cambiará sola de pantalla avisando que ha registrado un “PIÑASO”, pero te permite la opción de avisar de que ha sido una falsa alarma o que tu estado es bueno.





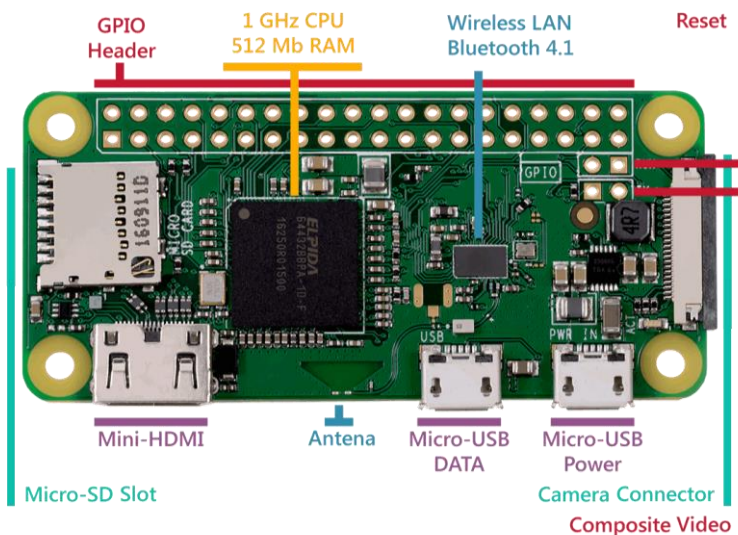
3.2.1. Diseño Arquitectónico



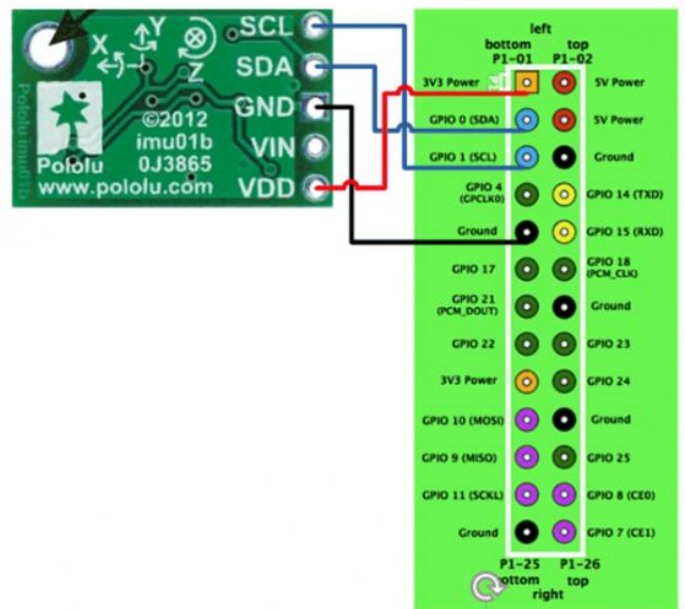
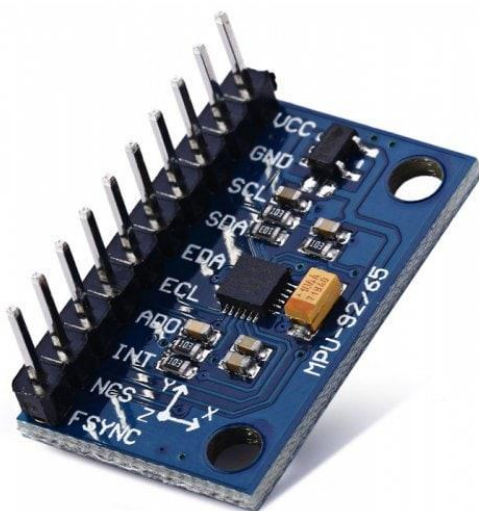


3.3. Dispositivo

El dispositivo esta soportado en una Raspberry Pi Zero WH, siendo esta la mas pequeña entre su familia.



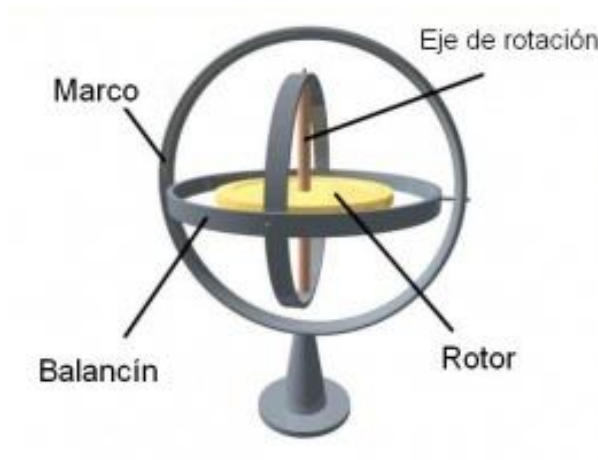
Tiene acoplado un módulo MPU-9250/6200, el cual es un dispositivo que tiene integrado un giroscopio, acelerómetro y un magnómetro.



En nuestro caso hemos usado el giroscopio, esto es un sensor mecánico que sirve para medir, mantener o cambiar la orientación en el espacio de algún aparato o vehículo. Está formado esencialmente por un cuerpo con simetría de rotación que gira alrededor del eje de dicha simetría.



Estos sensores miden la velocidad angular. Las unidades de medida de la velocidad angular es revoluciones por segundo (RPS).



Para entender cuantitativamente el movimiento de un gir6scopo, podemos utilizar la segunda ley de Newton para la rotaci3n.

$$d\mathbf{L} = \tau_{\text{neto}} dt$$

Junto con las relaciones.

$$\tau_{\text{neto}} = \mathbf{r} \cdot \mathbf{Mg}$$

$$\mathbf{L} = \mathbf{I}_s \omega_s$$

Donde I es el momento de inercia y ω es la velocidad angular de la rueda respecto a su eje de spin.

En un giroscopio debemos tener en cuenta que el cambio en el momento angular de la rueda debe darse en la direcci3n del momento de la fuerza que actúa sobre la rueda.

La velocidad angular de precesi3n puede calcularse de la siguiente manera:

En un peque1o intervalo de tiempo dt , el cambio experimentado por el momento angular tiene m3dulo dL :

$$d\mathbf{L} = \tau dt = MgD dt$$

En donde MgD es el m3dulo del momento respecto al punto donde pivota. El ángulo θ barrido por el eje en su movimiento es:

$$d\theta = \frac{d\mathbf{L}}{\mathbf{L}} = MgD \frac{dt}{\mathbf{L}}$$



Tras todo esto, se tiene que preparar un programa en Python para que este comprobando el estado del giroscopio.

En primer lugar, escribiremos el Shebang para obligar al sistema que lo ejecute con Python3, seguido importamos todas las librerías necesarias para su correcto funcionamiento.

```
#!/usr/bin/python3
import FaBo9Axis_MPU9250
import time
import sys
import shlex, subprocess
from signal import pause
```

Asignaremos una variable al atributo del objeto importado.

```
mpu9250 = FaBo9Axis_MPU9250.MPU9250()
```

Creemos un bucle para que esté en todo momento comprobando el estado del módulo. Comprobando los ejes X, Y, y Z. En cuanto se activen los controles establecidos, medidos en RPS, se enviará a la App la señal de control “pinaso”.

```
try:
    while True:
        gyro = mpu9250.readGyro()
        gx = float ( gyro['x'] )
        gy = float ( gyro['y'] )
        gz = float ( gyro['z'] )
        cont = int(0)
        print(gx,gy,gz)
        if ( gx > 180 or gx < -180 ):
            cont += 1
        if ( gy > 180 or gy < -180 ):
            cont += 1
        if ( gz > 180 or gz < -118 ):
            cont += 1
        if cont >= 2:
            subprocess.call(shlex.split('sudo echo "pinaso">/dev/rfcomm0'))
            archivo = open ("/dev/rfcomm0", "w")
            archivo.write("pinaso")
            archivo.close()
            exit()
        time.sleep(0.5)
```



El programa se ejecuta desde un Script en Bash para poder centralizar todas las acciones desde un mismo lugar. A parte, se crea un dispositivo virtual desde se realizará todo el envío de información. Este dispositivo es un interfaz Bluetooth.

```
#!/bin/bash

cd /
sudo rfcomm watch hci0 1 getty rfcomm0 1151200 vt100 -a pi &
cd /home/pi/Desktop/Pinaso/
sudo python3 pinaso.py
cd /
```

```
sudo rfcomm watch hci0 1 getty rfcomm0 1151200 vt100 -a pi &
```

Para poder crear este dispositivo denominado “rfcomm0” se tiene que configurar el fichero:

```
Fichero: /lib/systemd/system/bluetooth.service
```

Y escribir en él:

```
[Service]
Type=dbus
BusName=org.bluez
ExecStart=/usr/lib/bluetooth/bluetoothd -C
ExecStartPost=/usr/bin/sdptool add SP
ExecStartPost=/bin/hciconfig hci0 piscan
NotifyAccess=main
```

El Script Bash se ejecuta cada vez que el Piñaso se reinicia, eso se consigue gracias al editar en fichero:

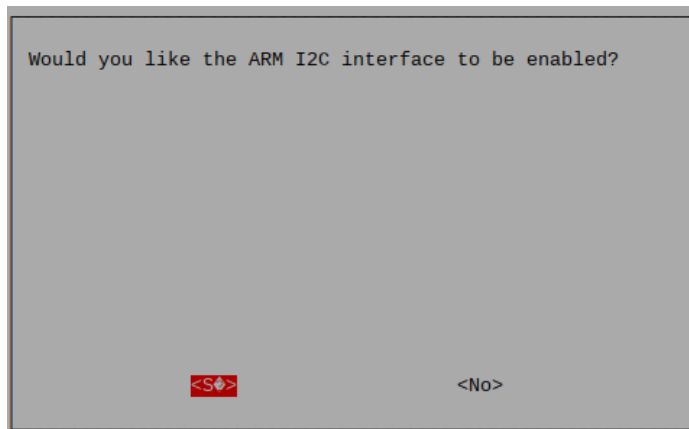
```
Fichero: /tmp/crontab.gFyLdW/crontab
```

Y escribir en él, al final de este mismo:

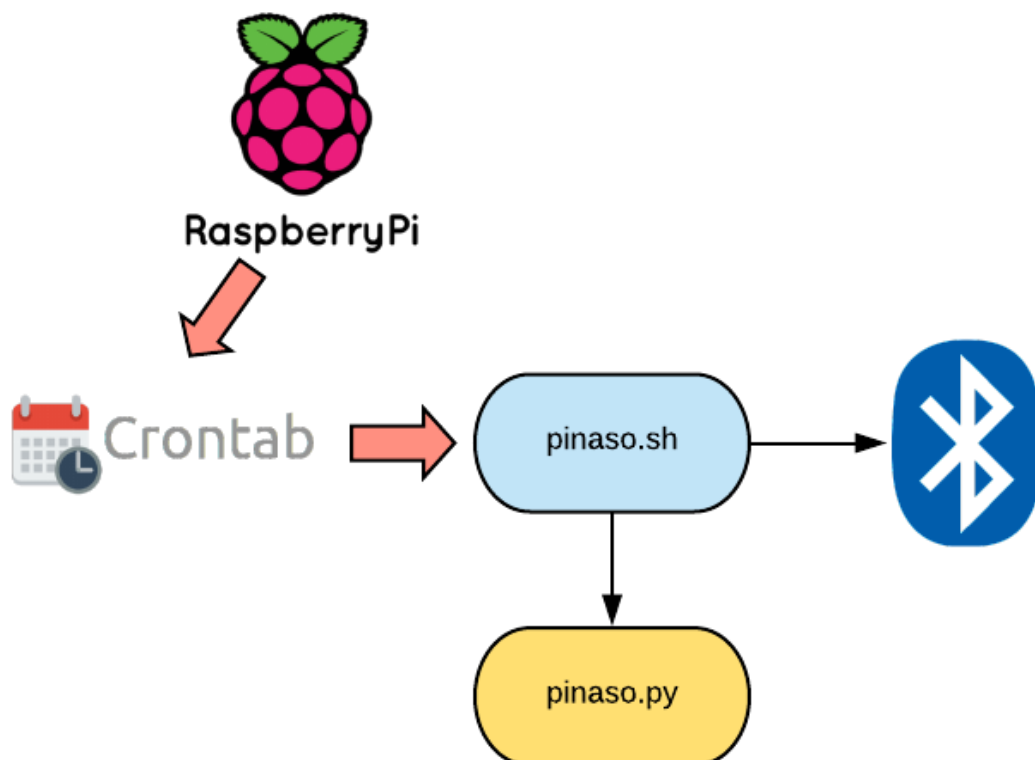
```
@reboot sudo bash /home/pi/Desktop/Pinaso/pinaso.sh
```



Para que la Raspberry se pueda comunicar con el módulo MPU-9250 se precisa de activar la comunicación Serial por el protocolo I2C.



3.3.1. Diseño Arquitectónico






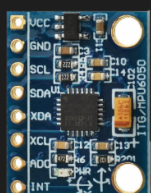
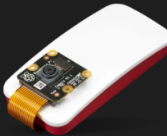
3.4. Web

La función principal de la página web es tener un portal donde poder adquirir el dispositivo, ya sea en cualquiera de sus modalidades, KIT o DIY (Do It Yourself).

La página esta desarrolla desde cero en HTML5, CSS3, Bootstrap, JavaScript, PHP7 y JQuery, estando soportada en AppServ. La idea era crear una página sencilla y bonita para el usuario.

desde la página se puede ver todos los porductos, pero se podrá comprar en caso de estar registrado y logeado.

Listado completo de los Productos:

	Raspberry Pi Zero W 15 € Añadir al Carrito
	MPU-9250/6500 15 € Añadir al Carrito
	Kit Pinaso 80 € Añadir al Carrito

Cualquier usuario puede registrarse sin problema.

Sign-Up

Nombre	<input type="text" value="Nombre"/>
Email	<input type="text" value="Email"/>
Password	<input type="password" value="Password"/>
Repeat Password	<input type="password" value="Password"/>
Type User	<input type="text"/>
<input type="button" value="Sign-Up"/> <input type="button" value="Return"/>	



Una vez realizado el Sign-Up, el quedará registrado en la base de datos MySQL y ya podrá acceder en cualquier momento mediante el Log-In.



Log-In

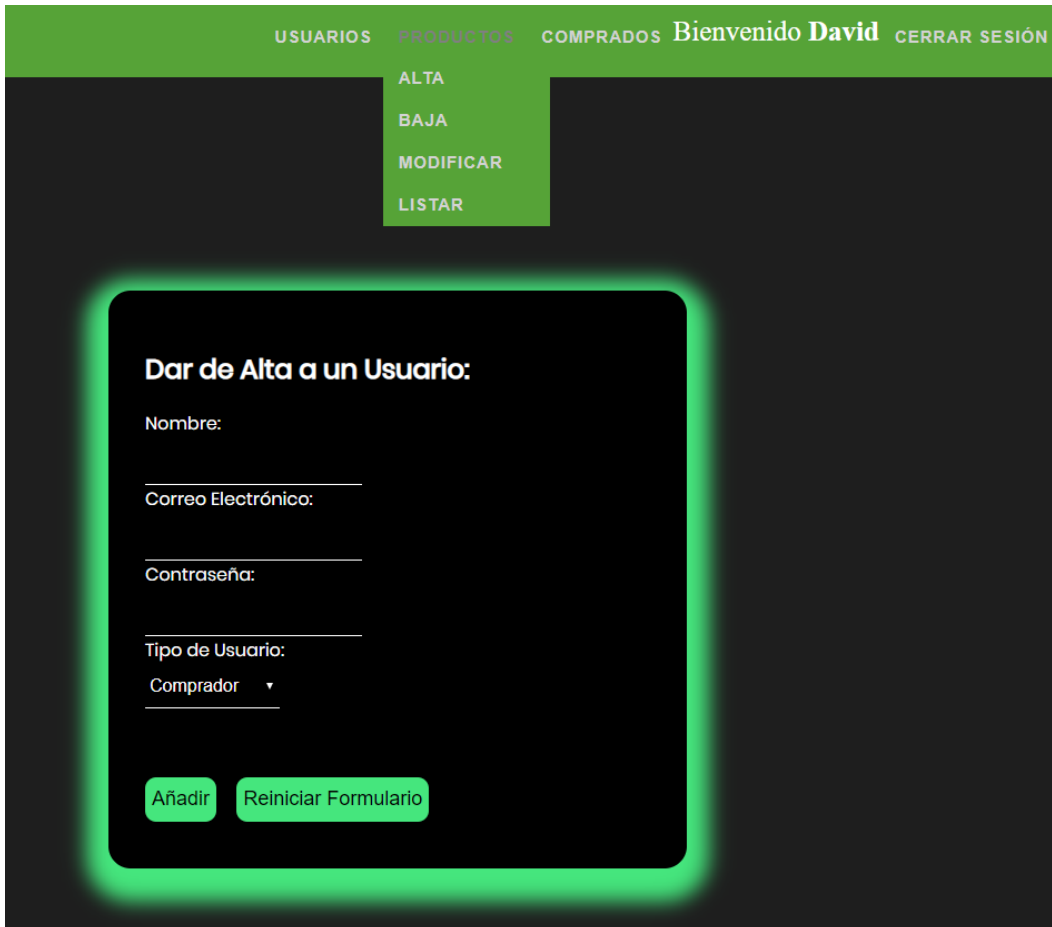
Email

Password

Al acceder el detectara quien es y le dará una bienvenida a cada usuario en la propia interfaz del menú de navegación.



Existen tres tipos de usuarios, Comprador (usuario habitual), Vendedor (proveedores de Piñaso) y los Admin. Estos últimos pueden realizar todo tipo de acciones administrando así la página.



USUARIOS PRODUCTOS COMPRADOS Bienvenido David CERRAR SESIÓN

ALTA
BAJA
MODIFICAR
LISTAR

Dar de Alta a un Usuario:

Nombre:

Correo Electrónico:

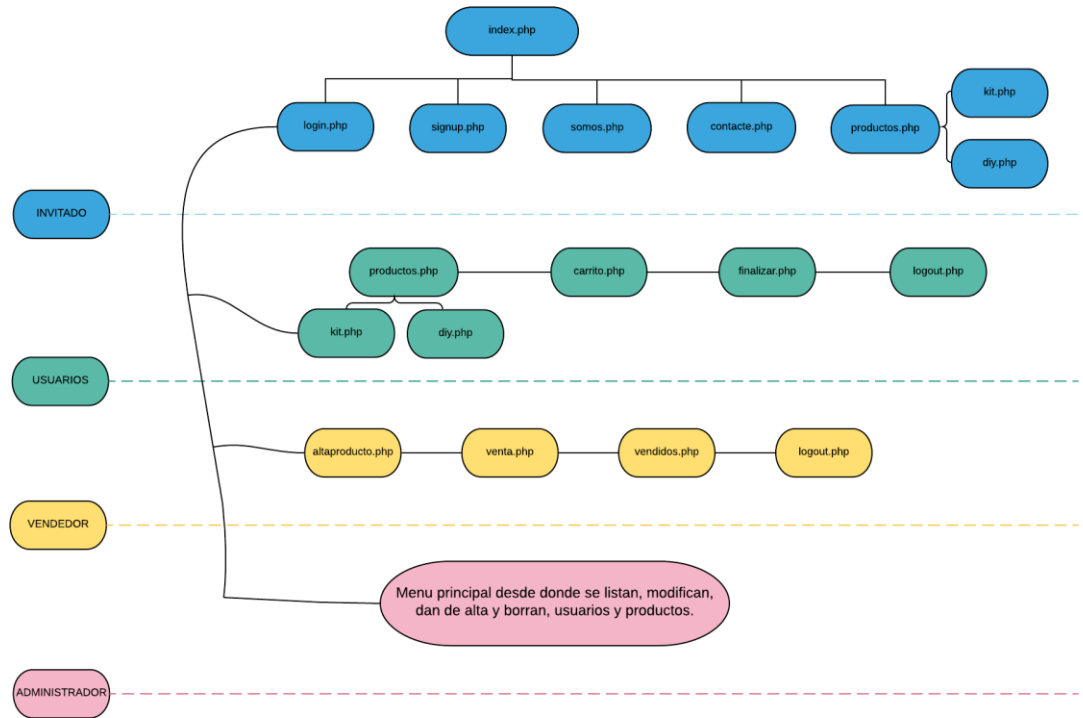
Contraseña:

Tipo de Usuario:
Comprador ▾

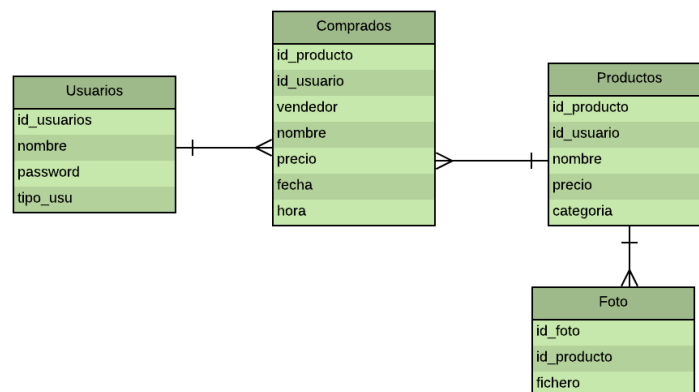


3.4.1. Diseño Arquitectónico

- WEB:**



- BBDD:**





4. Conclusión.

Una vez realizado el proyecto, estamos muy contentos con el resultado. Hemos sido capaces de desarrollar algo completamente desde cero y casi sin soporte, ya que algo como lo nuestro no está actualmente en el mercado. Estamos muy contentos con las capacidades autodidácticas que hemos ido desarrollando, así como nuestra capacidad de trabajo, donde nos hemos quedado en el centro todos los lunes y miércoles del curso alrededor de unas 4h.

Al principio, nosotros teníamos pensado en hacer otro proyecto completamente distinto, pero al vernos completamente desbordados con esa idea, se desistió y elegimos esta “Piñaso”, la cual es una idea más arriesgada, pero mucho más bonita e interesante. Cuando ya empezamos a desarrollar Piñaso nos dimos cuenta de que no existía información como hacerlo y tras mucho pensar, decidimos comprar un giroscopio y “perder” un par de días en calibrar cuando saltan las alertas. Cuando terminamos de tener el Python listo, nos pusimos con el desarrollo de la App, pero ninguno de los dos habíamos hecho antes una App, así que nos pusimos de nuevo a investigar y aprender como hacer uso de Kodular. Nuestra última y más dura “muro” fue el conectar la Raspberry con el Bluetooth, ya que nos estaba dando problemas por todos los lados, finalmente en un foro encontramos una manera de crear un dispositivo virtual en donde nos conectaríamos y mandaríamos líneas de texto a la App donde ahí ya terminaría todo el proceso de actuación.

Como resumen y conclusión, podemos afirmar que nuestro proyecto ha sido un proyecto de investigación, donde la mayor parte del tiempo hemos leído foros, preguntado profesionales, viendo videos y analizando todos nuestros problemas. Estamos muy contentos de como ha ido todo el desarrollo de Piñaso y la visión a futuro que puede llegar a tener el producto.



5. Bibliografía.

5.1. Módulo MPU-9250.

<https://www.youtube.com/watch?v=mzwovYcozvl&t=17s>

<https://www.youtube.com/watch?v=ucxF5m1twG4>

<https://www.youtube.com/watch?v=awjXzXnRuwo>

<https://github.com/cityofeden/cosmic/tree/master/Sensors/MPU%209250>

<https://www.youtube.com/watch?v=xEGLYBQCBqg>

5.2. Bluetooth.

<https://gist.github.com/keithweaver/3d5dbf38074cee4250c7d9807510c7c3>

<http://www.bujarra.com/raspberry-pi-usos-de-bluetooth/>

<https://www.raspberrypi.org/forums/viewtopic.php?t=97294>

[https://wiki.archlinux.org/index.php/Bluetooth_\(Español\)](https://wiki.archlinux.org/index.php/Bluetooth_(Español))

5.3. Kodular.

<http://kio4.com/index.htm>

Daniel Fernández Manzanero.

