

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
ESCUELA DE SISTEMAS



DISERTACIÓN PREVIA A LA OBTENCION DEL TÍTULO DE
INGENIERÍA EN SISTEMAS Y COMPUTACIÓN

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

LUIS ANTONIO JACHO ZAMBRANO

DIRECTOR: ING. ALFREDO CALDERÓN

QUITO ABRIL, 2017

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Dedicatoria

Dedico esta tesis de grado a mi familia, quienes me apoyaron incondicionalmente durante el desarrollo de este proceso académico.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Agradecimientos

Agradezco a mi familia por apoyarme siempre y a todos los maestros que formaron de una manera u otra en mi formación profesional.

ÍNDICE DE CONTENIDOS

1.	MARCO TEÓRICO.....	1
1.1	RASPBERRY PI.....	1
1.1.1	Características de Hardware de Rasperry Pi.....	1
1.1.1.1	Raspberry Pi Modelo A.....	1
1.1.1.2	Raspberry Pi Modelo A+.....	3
1.1.1.3	Raspberry Pi Modelo B.....	4
1.1.1.4	Raspberry Pi Modelo B+.....	5
1.1.1.5	Raspberry Pi 2 Modelo B.....	6
1.1.2	Características de Software de Raspberry Pi.....	8
1.1.2.1	New Out Of Box Software (NOOBS).....	8
1.1.2.2	Aeros.....	9
1.1.2.3	Raspbian.....	9
1.1.2.4	OSMC (Open Source Media Center).....	9
1.1.2.5	Windows IoT (Internet of Things) Core.....	9
1.1.2.6	RISC OS.....	10
1.1.2.7	Pidora.....	10
1.2	PYTHON.....	11
1.2.1	Definición.....	11
1.2.1.1	IDLE.....	11
1.2.1.2	Variables.....	12
1.2.1.3	Comentarios.....	12
1.2.1.4	Declaración if.....	13
1.3	ANDROID.....	13
1.4	SENSOR DS18B20.....	14
1.4.1	Alimentación.....	14
1.4.1.1	Mediante el pin de datos.....	14
1.4.1.2	Mediante fuente externa.....	15
1.5	BLUETOOTH.....	15
1.5.1	Arquitectura.....	17
2.	ANÁLISIS DE REQUERIMIENTO DE LA APLICACIÓN.....	18
2.1	REQUERIMIENTO DE SOFTWARE.....	18

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

2.1.1	Instalación del software Raspbian	18
2.1.2	Configuración de Mit App Inventor 2	27
2.1.3	Asignación de Ip estática al raspberry pi	30
2.1.4	Software de control remoto.....	35
2.2	REQUERIMIENTO DE HARDWARE.....	40
2.2.1	Conexión del sensor de temperatura.....	40
2.2.2	Conexión del bluetooth hc-06.....	42
2.3	METODOLOGÍA EN EL DESARROLLO DE APLICACIONES MÓVILES.....	43
2.3.1	Metodologías ágiles	43
2.3.2	Metodología mobile development process spiral	44
3.	DESARROLLO E IMPLEMENTACIÓN DE LA APLICACIÓN.....	46
3.1	PROGRAMACIÓN EN PYTHON PARA LA ADQUISICIÓN DE LA TEMPERATURA.....	46
3.1.1	Acceso remoto al raspberry pi desde windows.....	46
3.1.2	Explicación de la programación en python para la adquisición de temperatura ambiente.....	47
3.1.3	Instrucciones para envío de datos de forma serial desde raspberry pi.....	57
3.2	CREACIÓN DE LA APLICACIÓN MÓVIL EN APP INVENTOR.....	58
3.2.1	Creación del diseño de la interface en app inventor	58
3.2.2	Programación de la aplicación móvil en app inventor.....	68
3.3	PRUEBAS DE FUNCIONAMIENTO DE LA APLICACIÓN MÓVIL PARA EL MONITOREO DE TEMPERATURA AMBIENTE.....	73
4.	CONCLUSIONES Y RECOMENDACIONES	76
4.1	Conclusiones	76
4.2	Recomendaciones.....	77
5.	REFERENCIAS.....	78
6.	ANEXOS	79

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

ÍNDICE DE IMÁGENES

Imagen 1.1 Tarjeta Raspberry Pi 1 modelo A.....	2
Imagen 1.2 Tarjeta Raspberry Pi 1 Modelo A+.....	3
Imagen 1.3 Tarjeta Raspberry Pi 1 Modelo B	4
Imagen 1.4 Tarjeta Raspberry Pi 1 Modelo B+	5
Imagen 1.5 Tarjeta Raspberry Pi 2 Modelo B	6
Imagen 1.6 Identificación de componentes de la tarjeta Raspberry Pi 2 modelo B	7
Imagen 1.7 Tarjeta Raspberry Pi 3 Modelo B	7
Imagen 1.8. Acceso a IDLE de Python.....	11
Imagen 1.15 Alimentación del DS18B20 mediante el pin de datos	14
Imagen 1.16 Alimentación del DS18B20 mediante fuente externa.....	15
Imagen 1.17 Pila de protocolos bluetooth.....	17
Imagen 2.1 Página oficial de descarga del software Raspbian	19
Imagen 2.2 Escritura de la imagen de Raspbian en el micro SD	19
Imagen 2.3 Herramienta raspi-config	20
Imagen 2.4 Mensaje de expansión de la tarjeta SD	20
Imagen 2.5 Habilidad del inicio automático del LXDE	21
Imagen 2.6 Opción para seleccionar lenguaje, zona horaria	21
Imagen 2.7 Opción para configurar el teclado.....	22
Imagen 2.8 Selección del teclado genérico.....	23
Imagen 2.9 Selección del teclado español	24
Imagen 2.10 Selección del país de origen del teclado	24
Imagen 2.11 Selección de disposición de teclas	25
Imagen 2.12 Selección de tecla AltGr por defecto	25
Imagen 2.13 Opciones avanzadas.....	25
Imagen 2.14 Acceso remote SSH	26
Imagen 2.15 Habilidad del acceso remote SSH.....	26
Imagen 2.16 Raspbian instalado	27
Imagen 2.17 Resultados de búsqueda de App Inventor	28

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY
PI, CON TECNOLOGÍA MÓVIL.”

Imagen 2.18	Página principal de App Inventor	28
Imagen 2.19	Solicitud de correo para acceso a App Inventor	29
Imagen 2.20	Interface de programación de Mit App Inventor 2	30
Imagen 2.22	Parámetros necesarios para la comunicación mediante SSH.....	31
Imagen 2.23.	Confirmación de la comunicación SSH.....	31
Imagen 2.24	Terminal para comunicación SSH	32
Imagen 2.25	Copia de respaldo del archive de red	32
Imagen 2.26	Comando para editar archivo de red	33
Imagen 2.27	Aplicación nano de Raspbian.....	34
Imagen 2.28	Comandos para guardar y salir de la aplicación nano.....	34
Imagen 2.29	Opción para modificar el archive de red.....	35
Imagen 2.30	Ventana de bienvenida a la instalación	36
Imagen 2.31	Ventana de licencia GNU	36
Imagen 2.32	Ventana de selección de tipo de instalación	37
Imagen 2.33	Ventana de personalización de la instalación	37
Imagen 2.34	No instalar Servidor TightVNC	38
Imagen 2.35	Ventana de tareas adicionales	38
Imagen 2.36	Ventana de inicio de instalación	39
Imagen 2.37	Ventana que indica instalación completa.....	39
Imagen 2.38	Interface de TightVNC Viewer.....	40
Imagen 2.39	Terminales del GPIO del Raspberry Pi.....	41
Imagen 2.40	Conexión del DS18B20 con el Raspberry Pi.....	42
Imagen 2.41	Conexión del HC-06 al Rasperry Pi.....	43
Imagen 3.1	Dirección IP del equipo remoto	46
Imagen 3.2	Ventana de Autenticación de acceso	47
Imagen 3.3	Escritorio remote de Raspbian	47
Imagen 3.4	Procedimiento para abrir Python 2	48
Imagen 3.5	IDLE de Python	49
Imagen 3.6	Creación de Nuevo programa	49
Imagen 3.7	Línea de comando para abrir el fichero config	50

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY
PI, CON TECNOLOGÍA MÓVIL.”

Imagen 3.8 Modificación del fichero config.txt	51
Imagen 3.9 Librerías necesarias para la adquisición de temperatura.....	51
Imagen 3.10 Inicialización de los pines GPIO.....	52
Imagen 3.11 Búsqueda del directorio del sensor	53
Imagen 3.12 Función para leer trama enviada por DS18B20.....	53
Imagen 3.13 Trama de datos enviada por el DS18B20	54
Imagen 3.14 Función read_temp().....	55
Imagen 3.15 Función principal del programa en Python.....	55
Imagen 3.16 Ejecución de programa en Python	56
Imagen 3.17 Valores de temperatura en el shell de Python	56
Imagen 3.18 Configuración de la comunicación serie en Python.....	57
Imagen 3.19 Envío de datos de temperatura en forma serial	58
Imagen 3.20 Creación nuevo proyecto en App Inventor	58
Imagen 3.21 Asignación de nombre al proyecto	59
Imagen 3.22 Interface del diseño de App Inventor.....	59
Imagen 3.23 Asignación de título al proyecto	60
Imagen 3.24 Título de la aplicación.....	60
Imagen 3.25 Modificación de la propiedad DispHorizontal e Icono.....	61
Imagen 3.26 Elemento disposición horizontal.....	61
Imagen 3.27 Selección y configuración del componente SelectorDeLista.....	62
Imagen 3.28 Selección y configuración del componente Botón.....	62
Imagen 3.29 Apariencia del componente Conectar y del Botón	63
Imagen 3.30 Imagen agregada al visor de la aplicación	64
Imagen 3.31 Características del componente DisposiciónHorizontal2	64
Imagen 3.32 Etiquetas para visualizar datos de la temperatura	65
Imagen 3.33 Ubicación de la disposición horizontal 4 en el visor	65
Imagen 3.34 Ubicación de la etiqueta que muestra el tipo de clima.....	66
Imagen 3.35 Interface de la aplicación para el monitoreo de temperatura	67
Imagen 3.36 Componentes no visibles de la aplicación	68
Imagen 3.37 Pantalla para programación en bloques	69

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY
PI, CON TECNOLOGÍA MÓVIL.”

Imagen 3.38 Evento que lista clientes Bluetooth emparejados.....	69
Imagen 3.39 Bloques de programa después de la selección	70
Imagen 3.40 Bloque que se ejecutan cada 500 ms.....	71
Imagen 3.41 Bloques de programa botón Desconectar	72
Imagen 3.42 Bloque de programa que finaliza la aplicación.....	73
Imagen 3.43 Generación aplicación.....	73
Imagen 3.44 Aplicación abierta sin conexión.....	74
Imagen 3.45 Aplicación mostrando valor de temperatura	75

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

ÍNDICE DE CUADROS

Cuadro 1.1 Características eléctricas Raspberry Pi 1 modelo A	2
Cuadro 1.2 Características eléctricas Raspberry Pi 1 modelo A+	4
Cuadro 1.3 Características eléctricas Raspberry Pi 1 modelo B.....	5
Cuadro 1.4 Características eléctricas Raspberry Pi modelo B+	6
Cuadro 1.5 Características eléctricas Raspberry Pi 2 modelo B.....	7
Cuadro 1.6 Características eléctricas Raspberry Pi 3 modelo B.....	8
Cuadro 1.7 Especificaciones Bluetooth	16
Cuadro 1.8 Ancho de banda Bluetooth.....	16
Cuadro 3.1 Propiedades de diseño del componente Conectar	63
Cuadro 3.2 Propiedades de diseño del componente Botóón.....	63

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

GLOSARIO

AROS		Jack	
Sistema operativo de código abierto	9	Conector de audio de señales analógicas	2
CPU		LINUX	
Unidad Central de Proceso.....	2	Sistema operativo más prominente de	
Debian		software libre	9
Sistema operativo GNU basado software		memoria RAM	
libre	9	Memoria de acceso aleatorio	1
Ethernet		Memoria principal de la computadora,	
Estándar de redes de área local para		donde residen programas y datos, sobre	
computadoras.	1	la que se pueden efectuar operaciones	
Tecnología de red de área local	1	de lectura y escritura	1
Fedora		MicroUSB	
Distribución Linux	11	Tecnología USB de menor tamaño	2
GPIO		Raspberry Pi	
Entrada/salida de propósito general	3	Ordenador de placa única utilizado es este	
GPU		proyecto.....	1
Unidad de procesamiento gráfico	2	RCA	
HDMI		Conector eléctrico audiovisual.....	2
Interfaz multimedia de alta definición	2	Zip	
IDLE		Formato de compresión sin pérdida.....	9
Ambiente de desarrollo y aprendizaje			
integrado	11		

MARCO TEÓRICO

1.1 RASPBERRY PI

La Raspberry Pi ¹ es un ordenador de tamaño de una tarjeta de crédito que se conecta a un televisor y a un teclado. Es un pequeño ordenador que puede ser utilizado en proyectos de electrónica, y para muchas de las cosas que hace el PC de escritorio, como hojas de cálculo, procesamiento de texto, navegación por internet, e incluso instalar juegos. También reproduce vídeo de alta definición (Fundacion Raspberry Pi, s.f.).

1.1.1 CARACTERÍSTICAS DE HARDWARE DE RASPBERRY PI

1.1.1.1 *Raspberry Pi Modelo A*

El modelo A es el modelo de menor especificaciones, tiene 256 MB de memoria RAM², un puerto USB y no tiene un puerto Ethernet.

Este modelo es comprado usualmente para proyectos integrados ya que no consta de muchos puertos y tampoco cuenta con un chip para Ethernet³, es el modelo más ligero y consume menos energía que el modelo B, el campo que más demanda este modelo concierne a la robótica, y también en proyectos en donde el peso y el bajo consumo energético son elementales, como por el ejemplo el desarrollo de globos aerostáticos. El modelo A en combinación con una conexión apropiada Wi-Fi se convierte en un dispositivo multimedia ideal para conectar a un televisor (Fundación Raspberry, Raspberry Pi 1 Modelo A, s.f.) .

¹ Raspberry Pi: Computador de placa única utilizado en este proyecto.

² Memoria RAM: Memoria de Acceso Aleatorio

³ Ethernet: Tecnología de red de área local.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

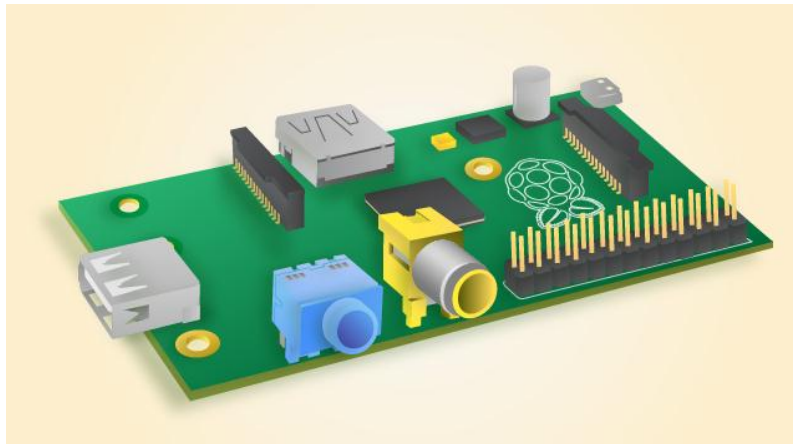


Imagen 1.1 Tarjeta Raspberry Pi 1 modelo A

Fuente: (Fundación Raspberry, Raspberry Pi 1 Modelo A, s.f.)

SoC (System-on-a-Chip)	Broadcom BCM2835
CPU ⁴	700 MHz ARM1176JZF-S
GPU ⁵	VideoCore IV
RAM	256 Mb
USB	1
VIDEO	RCA ⁶ , HDMI ⁷
AUDIO	Jack ⁸ , HDMI
BOOT	SD
RED	NO TIENE
CONSUMO	300 mA / 1.5w / 5v
ALIMENTACIÓN	MicroUSB ⁹ /GPIO ¹⁰
TAMAÑO	85,6 x 53,98 mm

Cuadro 1.1 Características eléctricas Raspberry Pi 1 modelo A

Fuente: (Fundación Raspberry, Raspberry Pi 1 Modelo A, s.f.)

⁴ CPU: Unidad Central de Proceso

⁵ GPU: Unidad de procesamiento gráfico

⁶ RCA: Conector eléctrico audiovisual

⁷ HDMI: interfaz multimedia de alta definición

⁸ Jack: Conector de audio de señales analógicas

⁹ MicroUSB: Tecnología USB de menor tamaño

¹⁰ GPIO: Entrada/Salida de propósito general

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Elaborado por: Luis Jacho

1.1.1.2 Raspberry Pi Modelo A+

El modelo A+ es una variación de bajo costo del modelo A, y lo reemplazó en noviembre del 2014, entre los cambios encontrados la cabecera GPIO tiene 40 pins, la integración de memoria Micro SD, consumo de energía optimizado, mejor audio con un circuito que cuenta con una fuente de alimentación de bajo ruido, y la placa es de tamaño menor, alrededor de 2 cm menos que el modelo A (Fundación Raspberry , Raspberry Pi 1 Modelo A+, s.f.).

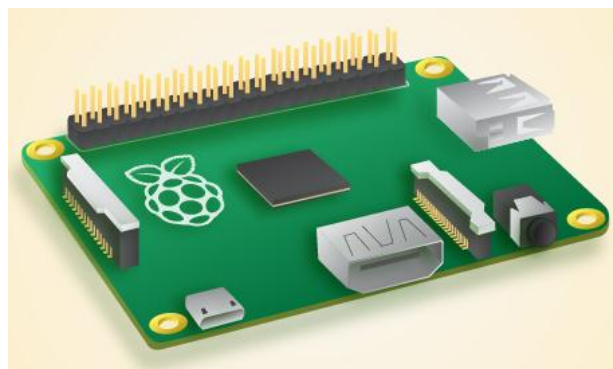


Imagen 1.2 Tarjeta Raspberry Pi 1 Modelo A+

Fuente: (Fundación Raspberry , Raspberry Pi 1 Modelo A+, s.f.)

SoC (System-on-a-Chip)	Broadcom BCM2835
CPU	700 MHz ARM1176JZF-S
GPU	VideoCore IV
RAM	256 Mb
USB	1
VIDEO	Jack, HDMI
AUDIO	Jack, HDMI
BOOT	Micro SD
RED	NO TIENE
CONSUMO	400 mA / 2w / 5v
ALIMENTACIÓN	MicroUSB /GPIO
TAMAÑO	65 x 56 mm

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Cuadro 1.2 Características eléctricas Raspberry Pi 1 modelo A+

Fuente: (Fundación Raspberry , Raspberry Pi 1 Modelo A+, s.f.)

Elaborado por: Luis Jacho

1.1.1.3 Raspberry Pi Modelo B

El modelo B es una variación con más altas especificaciones, tiene 512 MB de memoria RAM, dos puertos USB, y un puerto Ethernet de 100 Mb (Fundación Raspberry, Raspberry Pi 1 Modelo B, s.f.).

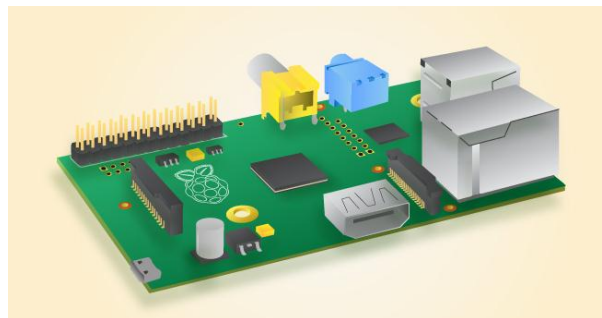


Imagen 1.3 Tarjeta Raspberry Pi 1 Modelo B

Fuente: (Fundación Raspberry, Raspberry Pi 1 Modelo B, s.f.)

SoC (System-on-a-Chip)	Broadcom BCM2835
CPU	700 MHz ARM1176JZF-S
GPU	VideoCore IV
RAM	512 Mb
USB	2
VIDEO	RCA, HDMI
AUDIO	Jack, HDMI
BOOT	Micro SD
RED	Ethernet 10/100
CONSUMO	700 mA / 3.5w / 5v
ALIMENTACIÓN	MicroUSB /GPIO
TAMAÑO	85,6 x 53,98 mm

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Cuadro 1.3 Características eléctricas Raspberry Pi 1 modelo B

Fuente: (Fundación Raspberry, Raspberry Pi 1 Modelo B, s.f.)

Elaborado por: Luis Jacho

1.1.1.4 Raspberry Pi Modelo B+

Reemplazó al modelo B en Julio de 2014, incluye cambios como el número de pins de la cabecera GPIO a 40, contiene 4 puertos USB 2.0 alineados a la placa, puerto micro SD de tipo push-push, y se añadieron cuatro agujeros de montaje colocados en ángulo recto (Fundación Raspberry, Raspberry Pi 1 Modelo B+, s.f.).

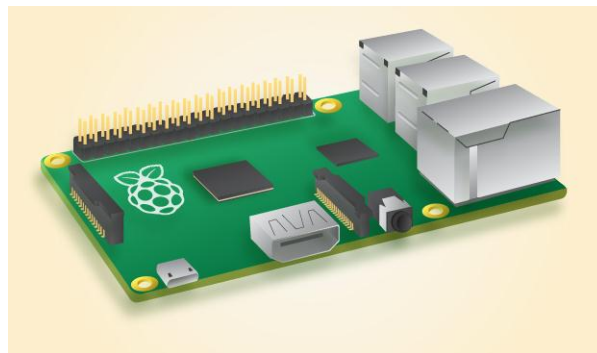


Imagen 1.4 Tarjeta Raspberry Pi 1 Modelo B+

Fuente: (Fundación Raspberry, Raspberry Pi 1 Modelo B+, s.f.)

SoC (System-on-a-Chip)	Broadcom BCM2835
CPU	700 MHz ARM1176JZF-S
GPU	VideoCore IV
RAM	512 Mb
USB	4
VIDEO	Jack, HDMI
AUDIO	Jack, HDMI
BOOT	Micro SD
RED	Ethernet 10/100
CONSUMO	500 mA / 2.5w / 5v
ALIMENTACIÓN	MicroUSB /GPIO
TAMAÑO	85x 56 mm

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Cuadro 1.4 Características eléctricas Raspberry Pi modelo B+

Fuente: (Fundación Raspberry, Raspberry Pi 1 Modelo B+, s.f.)

Elaborado por: Luis Jacho

1.1.1.5 Raspberry Pi 2 Modelo B

Constituye la segunda generación de Raspberry Pi, reemplazó al modelo Raspberry Pi 1 Modelo B+ en febrero de 2015, cuenta con la novedad de tener un SoC de 900MHz quad-core ARM Cortex – A7 CPU, y presenta una memoria RAM correspondiente a 1 GB (Fundación Raspberry, Raspberry Pi 2 Modelo B, s.f.).

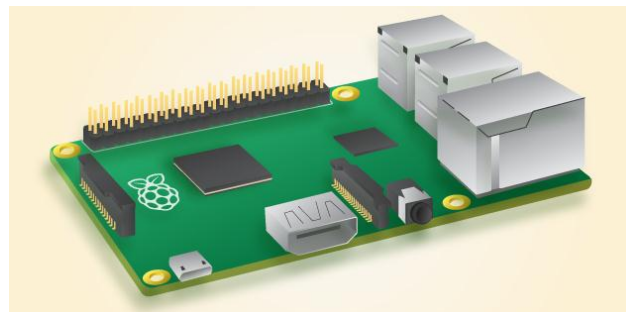


Imagen 1.5 Tarjeta Raspberry Pi 2 Modelo B

Fuente: (Fundación Raspberry, Raspberry Pi 2 Modelo B, s.f.)

SoC (System-on-a-Chip)	Broadcom BCM2836
CPU	900 MHz Quad-core-ARM Cortex-A7
GPU	VideoCore IV
RAM	1 GB
USB	4
VIDEO	Jack, HDMI
AUDIO	Jack, HDMI
BOOT	Micro SD
RED	Ethernet 10/100
CONSUMO	800 mA / 2.5w / 5v
ALIMENTACIÓN	MicroUSB /GPIO
TAMAÑO	85x 56 mm
PRECIO	35\$

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Cuadro 1.5 Características eléctricas Raspberry Pi 2 modelo B

Fuente: (Fundación Raspberry, Raspberry Pi 2 Modelo B, s.f.)

Elaborado por: Luis Jacho



Imagen 1.6 Identificación de componentes de la tarjeta Raspberry Pi 2 modelo B

Fuente: (Nergiza, s.f.)

1.1.3.7 Raspberry Pi 3 Modelo B

Constituye la tercera generación de Raspberry Pi, reemplazó a Raspberry Pi 2 Modelo B en febrero de 2016, con un procesador a 1.2GHz, conexión LAN, Bluetooth 4.1, y Bluetooth Low Energy (BLE) (Fundación Raspberry, Raspberry Pi 3 Modelo B, s.f.).

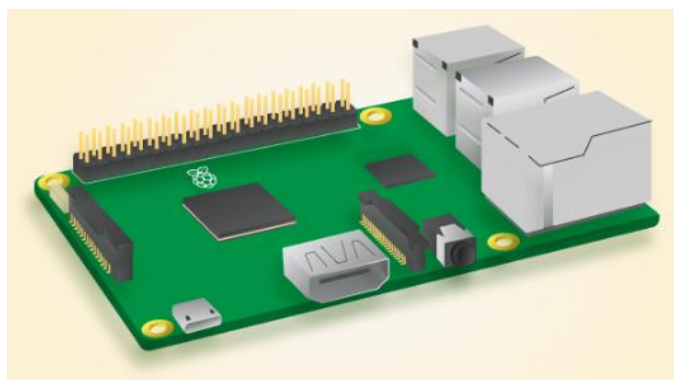


Imagen 1.7 Tarjeta Raspberry Pi 3 Modelo B

Fuente: (Fundación Raspberry, Raspberry Pi 3 Modelo B, s.f.)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

SoC (System-on-a-Chip)	Broadcom BCM2837
CPU	1.2Ghz ARM Cortex-A53
GPU	VideoCore IV
RAM	1GB
USB	4
VIDEO	Jack, HDMI
AUDIO	Jack, HDMI
BOOT	Micro SD
RED	Ethernet 10/100, Wifi, BT
CONSUMO	2,5 A / 12,5w / 5v
ALIMENTACIÓN	MicroUSB /GPIO
TAMAÑO	85x 56mm
PRECIO	35\$

Cuadro 1.6 Características eléctricas Raspberry Pi 3 modelo B

Fuente: (Fundación Raspberry, Raspberry Pi 3 Modelo B, s.f.)

Elaborado por: Luis Jacho

1.1.2 CARACTERÍSTICAS DE SOFTWARE DE RASPBERRY PI

1.1.2.1 *New Out Of Box Software (NOOBS)*

Este sistema operativo está desarrollado para los usuarios principiantes de Raspberry Pi. No se necesita conexión a internet, y tampoco se necesita ninguna descarga adicional de software. Para adquirirlo es necesario abrir el link de descargas de la página oficial, y elegir el botón con el logo de NOOBS:

Una vez descargado el archivo Zip¹¹ de NOOBS, debemos desempacararlo dentro de una memoria SD de 4 GB como mínimo. Cuando se arranca por primera vez, un menú se despliega dando la opción al usuario de elegir un sistema operativo en particular:

¹¹ ZIP: Formato de compresión sin pérdida

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Una vez instalado el sistema operativo, el dispositivo arrancará con normalidad, pero NOOBS quedará instalado en la tarjeta SD, así que si presionamos shift en el arranque se abrirá el modo recuperación de NOOBS.

1.1.2.2 Aeros

Aeros es una distribución híbrida entre AROS¹² y LINUX¹³, y es de acceso concedido solamente a usuarios registrados, y para registrarse se hace una compra o donación en su página web.

1.1.2.3 Raspbian

Es un sistema operativo gratuito basado en Debian¹⁴, optimizado para hardware Raspberry Pi, es decir provee una serie de utilidades y programas para que el dispositivo funcione correctamente. Además del puro sistema operativo, Raspbian lleva consigo alrededor de 35000 paquetes de software pre-compilados liados a un formato de fácil instalación para Raspberry Pi.

1.1.2.4 OSMC (Open Source Media Center)

Es una distribución de Linux minimalista, auto actualizable, y que lleva al reproductor multimedia Kodi como su interfaz principal. Este proyecto fue fundado por Sam Nazarko en el año 2014, y es mantenido por un grupo de voluntarios en su tiempo libre. Este sistema operativo está basado en Debian, lo que significa que tiene alrededor de 30000 paquetes disponibles en su repositorio, por lo que se convierte en un sistema operativo expandible. Es muy rápido de instalar brindando una experiencia HTPC (Home Theater Personal Computer) en cuestión de minutos. El proyecto recibe actualizaciones constantes por lo que mejora con el paso del tiempo, y el sistema operativo consta con un almacén de aplicaciones lo que hace que la adición de nuevas características sea muy simple.

1.1.2.5 Windows IoT (Internet of Things) Core

Es una versión de Windows 10 que se encuentra optimizada para dispositivos pequeños con o sin monitor, y está disponible para Raspberry Pi 2 y 3. Además de

¹² AROS: Sistema operativo de código abierto

¹³ LINUX: Sistema operativo más prominente de software libre

¹⁴ Debian: Sistema operativo GNU basado en software libre

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

brindar el poder de Windows al dispositivo, hace mucho más fácil la integración de experiencias ricas a los proyectos en los que se esté trabajando tales como interfaces de usuario más naturales e intuitivas, buscadores, almacenamiento online, y servicios basados en la nube. “Windows 10 IoT Core es una versión muy limitada de Windows 10 que no cuenta con muchas de las opciones de su hermano mayor porque, simplemente, no las necesita.” (Pastor, 2015).

1.1.2.6 RISC OS

RISC OS es un sistema operativo diseñado en Cambridge, Inglaterra, por Acorn Computers, empresa que desapareció en noviembre del 2000, lanzado por primera vez en 1987. No es un sistema operativo común, pero puede ser entretenido trabajar con el mismo, no está basado en Linux, y es mucho más sencillo que los sistemas operativos más comunes, una aplicación puede tomar el sistema entero, y solo funciona para un solo usuario. No es muy popular y tiene muchas desventajas frente a los sistemas operativos con los cuales compete.

1.1.2.7 Pidora

Pidora es una distribución Linux publicada recientemente por el equipo CDOT (Centre for Development of Open Technology) de Seneca College, para el computador de placa única Raspberry Pi. Contiene paquetes de software del sistema operativo llamado Fedora, compilados específicamente para la arquitectura ARMv6 usada en los dispositivos Raspberry Pi.

Pidora es un remix de Fedora¹⁵, esto quiere decir que puede incluir software de terceros en su arquitectura, dicho software puede ser desarrollado por cualquier miembro de la comunidad, con diferentes objetivos como por ejemplo: apuntar el proyecto hacia dispositivos de hardware específicos, ofrecer material promocional dentro de un sistema operativo personalizable a los consumidores, entre otros.

La última versión de este Remix de Fedora cuenta con la mayoría de paquetes con los que viene Fedora 18, y son instalables a través de yum, un paquete administrador

¹⁵ Fedora: Distribución Linux

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

de software utilizado para instalar, actualizar, y remover paquetes en distribuciones Linux que manejan paquetes.

1.2 PYTHON

1.2.1 DEFINICIÓN

Python es un lenguaje interpretado, que quiere decir que un “intérprete” va leyendo las instrucciones y ejecutándolas en tiempo real. Un intérprete es un software encargado de convertir nuestras sentencias a código máquina. Puedes instalar cualquiera de los muchos intérpretes que existen para tu máquina, ya sea tu Raspberry Pi u ordenador personal (Fabrica Digital, 2016).

1.2.1.1 IDLE¹⁶

La más fácil introducción a Python es a través de IDLE, un entorno de desarrollo Python. El IDLE se abre desde el escritorio o el menú de aplicaciones.



Imagen 1.8. Acceso a IDLE de Python

Fuente: (Fundación Raspberry, s.f.)

¹⁶ IDLE: Ambiente de desarrollo y aprendizaje integrado

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

IDLE da un REPL (Read Evaluate Print Loop), el cual es un símbolo que indica que se puede escribir comandos de Python. Un REPL incluso obtiene la salida de comandos impresos en la pantalla sin usar el comando *print*.

Están disponibles dos versiones de Python: Python 2 y Python 3. Python 3 es la versión más reciente, sin embargo Python 2 está disponible para aplicaciones que no soportan Python 3 todavía.

Puede utilizar variables si es necesario, pero incluso puede utilizarlo como una calculadora. Por ejemplo:

```
>>> 1 + 2
3
>>> name = "Sarah"
>>> "Hello " + name
'Hello Sarah'
```

1.2.1.2 Variables

Para guardar un valor en una variable, se asigna de la siguiente manera:

```
name = "Bob"
age = 15
```

Tenga en cuenta que no he asignado tipos a estas variables, ya que los tipos son inferidos, y se pueden cambiar (son dinámicos).

```
age = 15
age += 1      # increment age by 1
print(age)
```

Esta vez se usó un comentario junto al comando de increment.

1.2.1.3 Comentarios

Los comentarios se ignoran en el programa, pero sirven para dejar notas, y se indican con el símbolo numeral (#). Los comentarios de varias líneas usan comillas triples:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

```
"""
```

```
This is a very simple Python program that prints "Hello".
```

```
That's all it does.
```

```
"""
```

```
print("Hello")
```

1.2.1.4 Declaración if

Puede utilizar las instrucciones if para el flujo de control:

```
name = "Joe"
```

```
if len(name) > 3:
```

```
    print("Nice name,")
```

```
    print(name)
```

```
else:
```

```
    print("That's a short name,")
```

```
    print(name)
```

1.3 ANDROID

Para programar en Android, existen algunas maneras, algunas de las cuales son:

- Con **Android SDK y Eclipse**. Se realizan los programas escribiendo códigos en Java. Es la forma que utilizan los buenos programadores.
- Mediante una aplicación de pago llamada **(B4A) BASIC4Android**, para realizar programas de manera parecida a como se hace con Visual Basic
- Mediante App inventor. Es una forma intuitiva de hacer programas situando convenientemente las instrucciones como si fueran piezas de un rompecabezas.

1.4 SENSOR DS18B20

El DS18B20 es un sensor digital de temperatura que utiliza el protocolo 1-Wire para comunicarse, este protocolo necesita solo un pin de datos para comunicarse y permite conectar más de un sensor en el mismo bus (naylampmechatronics, s.f.).

El sensor DS18B20 es fabricado por Maxim Integrated, el encapsulado de fábrica es tipo TO-92 similar al empleado en transistores pequeños. La presentación comercial más utilizada por conveniencia y robustez es la del sensor dentro de un tubo de acero inoxidable resistente al agua.

Con este sensor podemos medir temperatura desde los -55°C hasta los 125°C y con una resolución programable desde 9 bits hasta 12 bits.

Cada sensor tiene una dirección única de 64bits establecida de fábrica, esta dirección sirve para identificar al dispositivo con el que se está comunicando, puesto que en un bus 1-wire pueden existir más de un dispositivo.

1.4.1 ALIMENTACIÓN

1.4.1.1 Mediante el pin de datos

De esta forma, el sensor internamente obtiene energía del pin de datos cuando este se encuentra en un estado alto y almacena carga en un condensador para cuando la línea de datos esté en un estado bajo, a esta forma de obtener energía se le llama “Parasite Power” y se usa cuando el sensor debe conectarse a grandes distancias o en donde el espacio es limitado, puesto que de esta forma no se necesita la línea de VDD. El diagrama para su conexión debe ser de la siguiente forma:

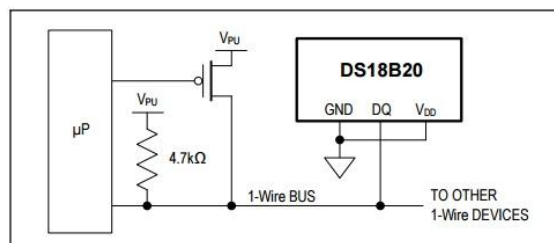


Imagen 1.9 Alimentación del DS18B20 mediante el pin de datos

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Fuente: (naylampmechatronics, s.f.)

Note que el pin GND y VDD están ambos conectados a GND, esto es indispensable para que se active el Parasite Power. El MOSFET en la imagen es necesario para cuando se realicen conversiones de temperatura o copiar datos desde la memoria de circuito de la EEPROM, en estas operaciones la corriente de operación aumenta y si solo se suministra energía a través de la resistencia pueden causar caídas de voltaje en el condensador interno (naylampmechatronics, s.f.).

1.4.1.2 Mediante fuente externa

De esta forma el sensor se alimenta a través del pin VDD, de esta forma el voltaje es estable e independiente del tráfico del bus 1-wire.

El diagrama de conexión es de la siguiente forma:

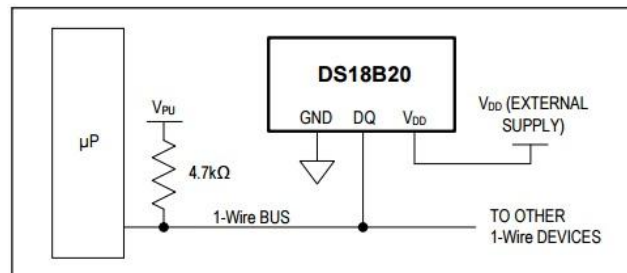


Imagen 1.10 Alimentación del DS18B20 mediante fuente externa

Fuente: (naylampmechatronics, s.f.)

1.5 BLUETOOTH

Bluetooth es una especificación industrial que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz. ISM (Industrial, Scientific and Medical) son bandas reservadas internacionalmente para uso no comercial de radiofrecuencia electromagnética en áreas industrial, científica y médica (Vega, 2014).

Los principales objetivos que se pretenden conseguir con esta norma son: facilitar las comunicaciones entre equipos móviles y fijos, eliminar los cables y conectores entre éstos, ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales. Bluetooth es un protocolo de comunicaciones diseñado especialmente

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPERRY PI, CON TECNOLOGÍA MÓVIL.”

para dispositivos de bajo consumo de energía eléctrica, con una cobertura baja y basados en circuitos integrados de bajo costo (Vega, 2014).

Los dispositivos bluetooth se clasifican en tres clases: "Clase 1", "Clase 2" y "Clase 3", y es únicamente una referencia de la potencia de transmisión del dispositivo, siendo totalmente compatibles los dispositivos de una clase con los de la otra (Vega, 2014).

El cuadro 1.7 muestra las especificaciones del protocolo bluetooth.

CLASE	Potencia máxima permitida (mW)	Potencia máxima permitida (dBm)	RANGO (aproximado)
Clase 1	100	20	100 m
Clase 2	2.5	4	10 m
Clase 3	1	0	1 m

Cuadro 1.7 Especificaciones Bluetooth

Fuente: (Vega, 2014)

El cuadro 1.8 presenta la versión y el ancho de banda del bluetooth.

Versión	Ancho de Banda
1.2	1 Mbps
2.0+EDR	3 Mbps
UWB	53 – 480 Mbps

Cuadro 1.8 Ancho de banda Bluetooth

Fuente: (Vega, 2014)

Los dispositivos que con mayor frecuencia utilizan esta tecnología pertenecen a sectores de las telecomunicaciones y la informática personal.

1.5.1 ARQUITECTURA

La especificación principal de Bluetooth (denominada core) define el nivel físico (PHY) y el control de acceso al medio (MAC). La especificación principal cubre los cuatro niveles inferiores y sus protocolos asociados junto con el protocolo de descubrimiento de servicios (SDP) y el perfil de acceso genérico (Castellano, 2012).

Los niveles inferiores de la pila de protocolos constituyen el controlador Bluetooth (hardware), que contiene los bloques fundamentales de la tecnología, sobre los cuales se apoyan los niveles superiores y los protocolos de aplicación. El controlador Bluetooth se compone de:

- El nivel de radiofrecuencia.
- El nivel de Banda Base.
- El nivel de Gestión de Enlace

El resto de niveles de base y los protocolos de aplicación residen en el anfitrión Bluetooth, o Host y se definen a nivel software. El Host se comunica con el controlador utilizando un interfaz estándar. Ambas entidades pueden integrarse para su uso conjunto en sistemas embebidos, o se pueden utilizar de forma intercambiable. El nivel más importante del anfitrión es el protocolo de control y adaptación de enlace lógico o L2CAP (Castellano, 2012).

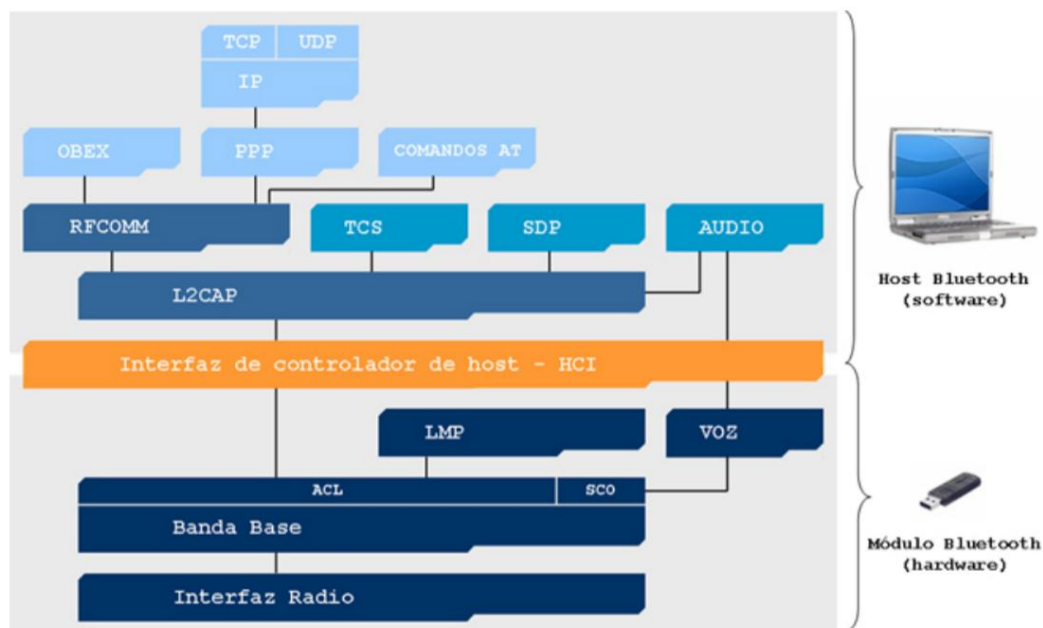


Imagen 1.11 Pila de protocolos bluetooth

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Fuente: (Castellano, 2012)

Los 4 niveles arriba descritos no son imprescindibles para que un sistema Bluetooth funcione, aunque son los que se definen como obligatorios. Es posible que se necesite una capa HCI que permite disponer de una capa de acceso homogénea para todos los módulos Bluetooth de banda base, aunque sean de distintos fabricantes.

ANÁLISIS DE REQUERIMIENTO DE LA APLICACIÓN

2.1 REQUERIMIENTO DE SOFTWARE

2.1.1 INSTALACIÓN DEL SOFTWARE RASPBIAN

Para la instalación del software Raspbian, es necesario disponer de los siguientes dispositivos:

- Raspberry Pi 2 Modelo B V1.1
- Tarjeta SD card de 4 GB o superior
- Teclado USB
- Mouse USB
- Cable HDMI
- Un Televisor con entrada HDMI
- Fuente de 5Vdc a 2 amperios

Luego descargue el software del sitio oficial <https://www.raspberrypi.org/downloads/>

En esta página pulse sobre el icono RASPBIAN, este le direccionará a una página donde se encuentra la última versión de este software, descargue el que tiene la extensión ZIP.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

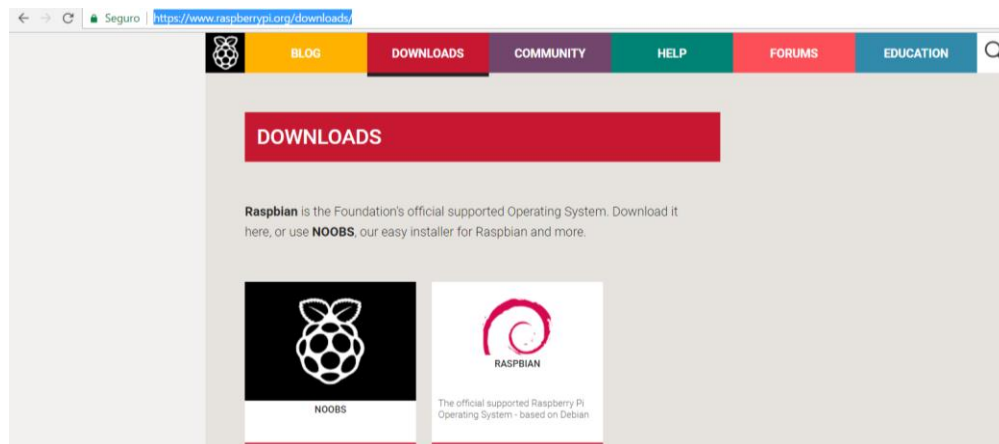


Imagen 2.1 Página oficial de descarga del software Raspbian

Fuente: (Fundación Raspberry, Downloads, 2017)

Una vez descargado el software Raspbian, descomprímalo en una ruta específica.

Descargue utilidad Win32DiskImager de la [página del proyecto en SourceForge](#), luego instale.

Con la ayuda de un adaptador inserte la micro SD en el computador, verifique la letra que Windows le asigna a esta unidad, por ejemplo (G:). Ejecute Win32DiskImager, seleccione la imagen del software Raspbian que se descomprimió, Haga clic Write y espere a que la escritura se complete.

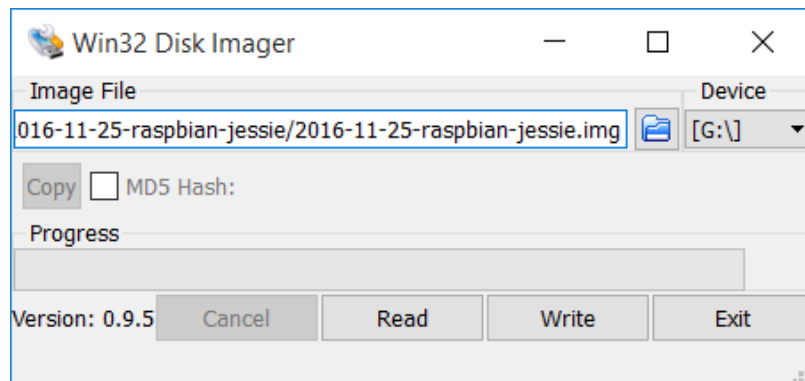


Imagen 2.2 Escritura de la imagen de Raspbian en el micro SD

Fuente: (Win32DiskImager, 2017)

Salga de Win32DiskImager y expulse el micro SD.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Conecte el teclado y mouse en los puertos USB del Raspberry Pi, el micro SD conecte en su respectiva ranura, con la ayuda del cable HDMI conecte la Raspberry a la televisión que debe estar configurada como entrada de video HDMI, para que sirva como interface y finalmente energice el microcomputador Raspberry Pi.

Luego del proceso de inicio del software Raspbian, aparece la herramienta raspi-config, esta se emplea para configura el dispositivo.

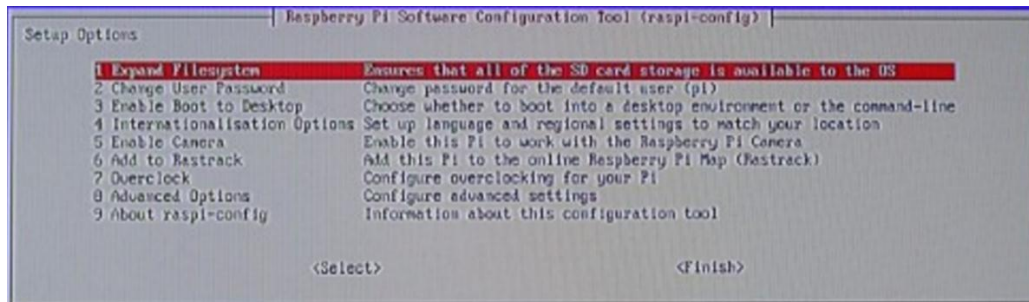


Imagen 2.3 Herramienta raspi-config

Fuente: (Raspbian, 2017)

La primera opción consiste en expandir el filesystem para que ocupe toda la capacidad de la tarjeta SD. Al pulsar enter desde el teclado, aparece un mensaje donde menciona que el proceso de expansión se realizará en el próximo reinicio del microcomputador Raspberry.

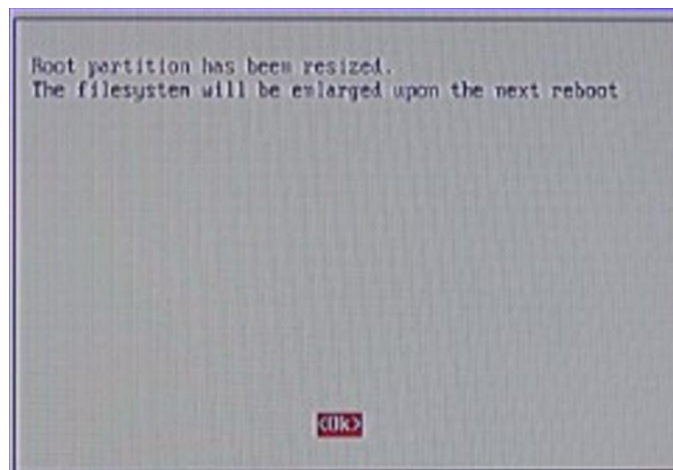


Imagen 2.4 Mensaje de expansión de la tarjeta SD

Fuente: (Raspbian, 2017)

La segunda opción sirve para cambiar la contraseña de usuario, para este caso se mantendrá la contraseña de fábrica.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

La tercera opción habilita el inicio automático al escritorio LXDE (Lightweight X11 Desktop Environment), es un entorno de **escritorio** extremadamente rápido, eficiente y de bajo consumo. Para este caso no se seleccionó esta opción.

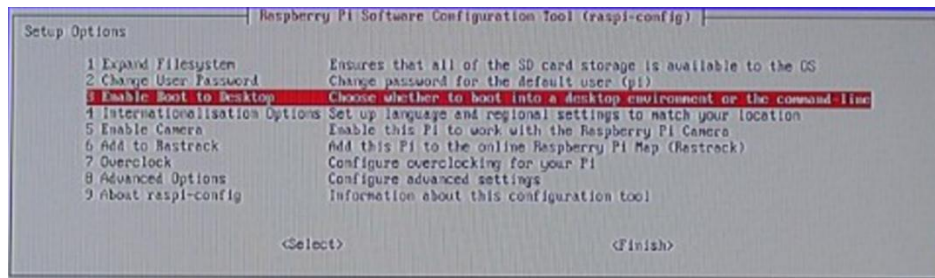


Imagen 2.5 Habilidad del inicio automático del LXDE

Fuente: (Raspbian, 2017)

Mediante la opción cuatro que es "Internationalisation options" es posible configurar el lenguaje, zona horaria y teclado. Se seleccionó para este caso el lenguaje Español-Ecuador y la zona horaria América-Guayaquil.

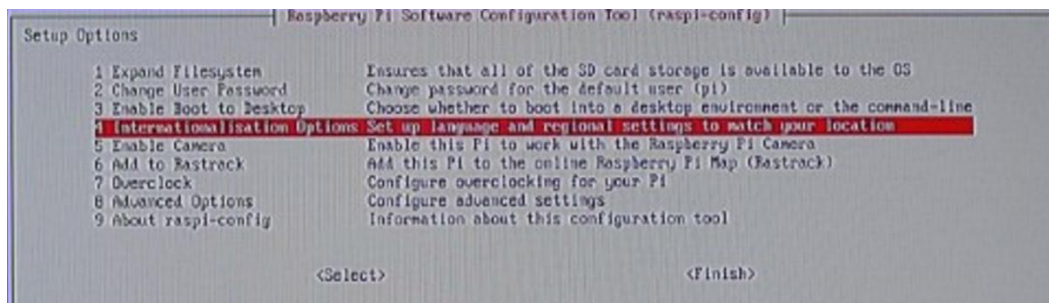


Imagen 2.6 Opción para seleccionar lenguaje, zona horaria

Fuente: (Raspbian, 2017)

Para configurar el teclado seleccione la opción que corresponde:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

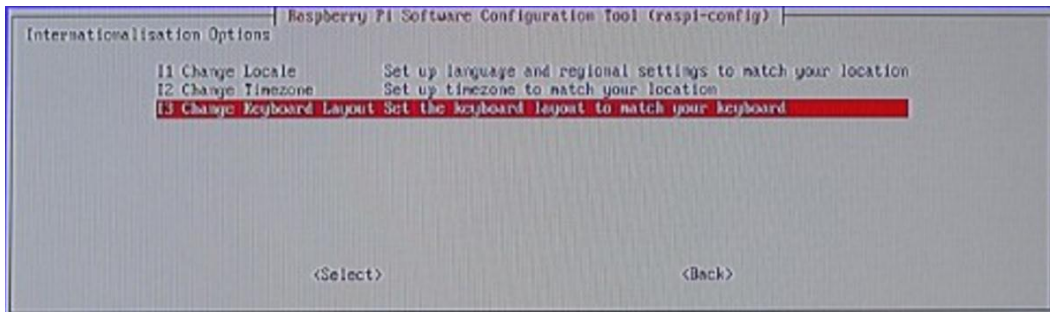


Imagen 2.7 Opción para configurar el teclado

Fuente: (Raspbian, 2017)

Seleccione el modelo de teclado genérico:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

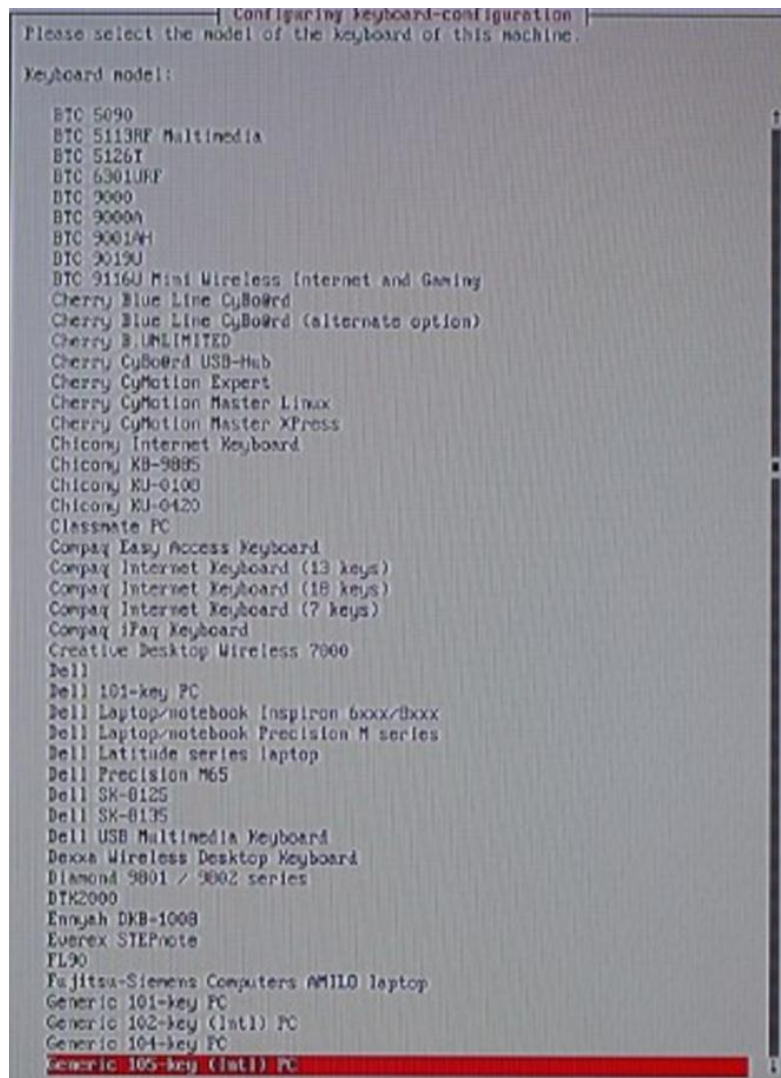


Imagen 2.8 Selección del teclado genérico

Fuente: (Raspbian, 2017)

Como el teclado utilizado es español seleccione la disposición (layout) "Other":

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

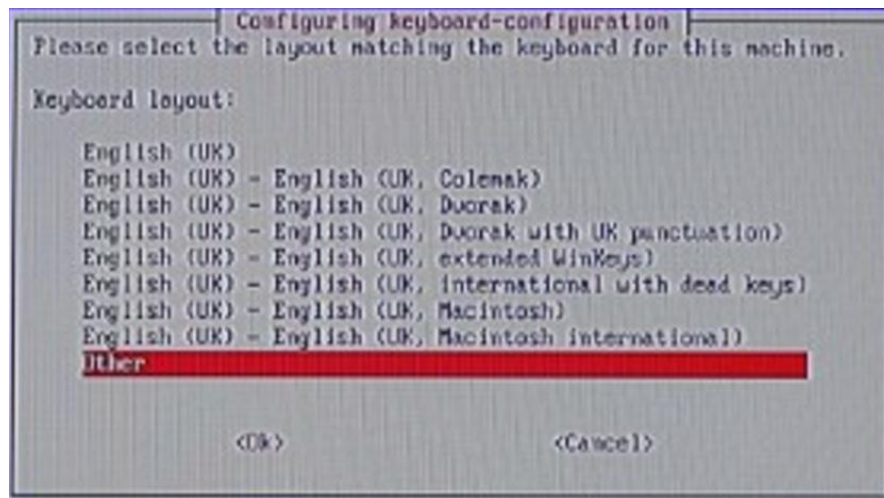


Imagen 2.9 Selección del teclado español

Fuente: (Raspbian, 2017)

Luego seleccione el país de origen Spanish (Latin American):



Imagen 2.10 Selección del país de origen del teclado

Fuente: (Raspbian, 2017)

En disposición de teclas también seleccionamos Spanish (Latin American):

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

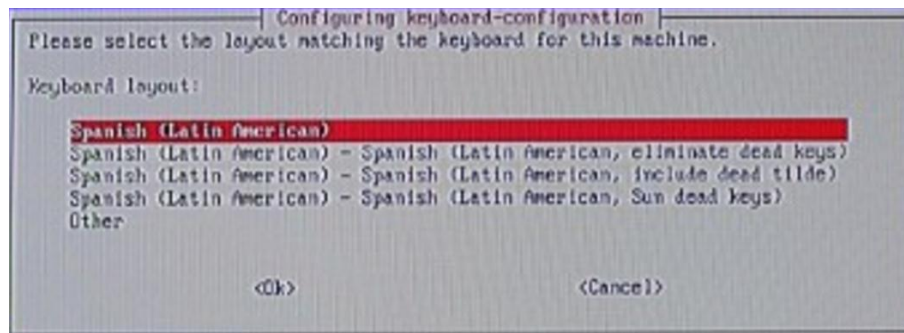


Imagen 2.11 Selección de disposición de teclas

Fuente: (Raspbian, 2017)

La tecla AltGr, seleccionamos la que viene predefinida en el teclado:

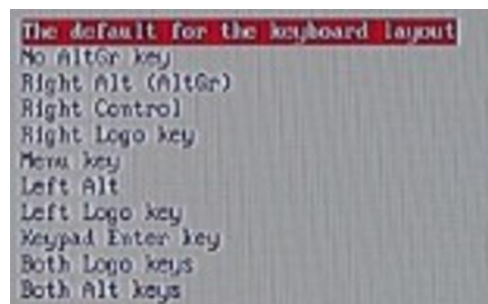


Imagen 2.12 Selección de tecla AltGr por defecto

Fuente: (Raspbian, 2017)

En la opción 8 que corresponde a Advanced Options es posible habilitar el servidor SSH, necesario para comunicación entre dispositivos:

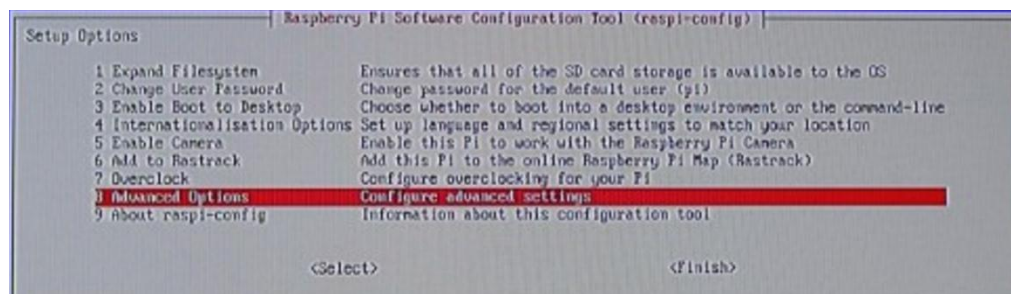


Imagen 2.13 Opciones avanzadas

Fuente: (Raspbian, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

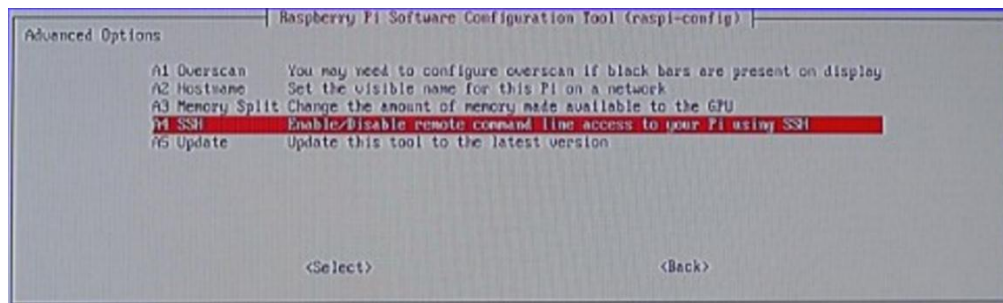


Imagen 2.14 Acceso remote SSH

Fuente: (Raspbian, 2017)

Cuando aparezca una ventana donde pregunta si le gustaría habilitar o deshabilitar el servidor SSH, seleccione Enable, pulse enter, aparece un mensaje donde indica que el servidor ya está habilitado:

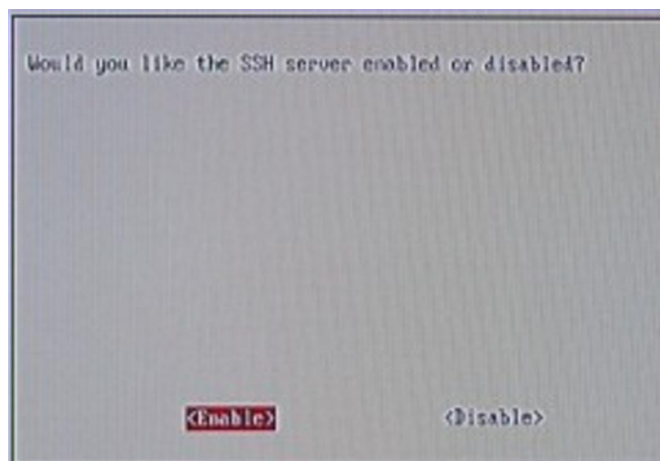


Imagen 2.15 Habilitación del acceso remote SSH

Fuente: (Raspbian, 2017)

Realizado estos cambios en la configuración del Raspberry Pi, seleccione Finish para que el microcomputador se reinicie.

Después del reinicio la interface del Raspbian instalado es como la que se muestra en la imagen 2.16.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

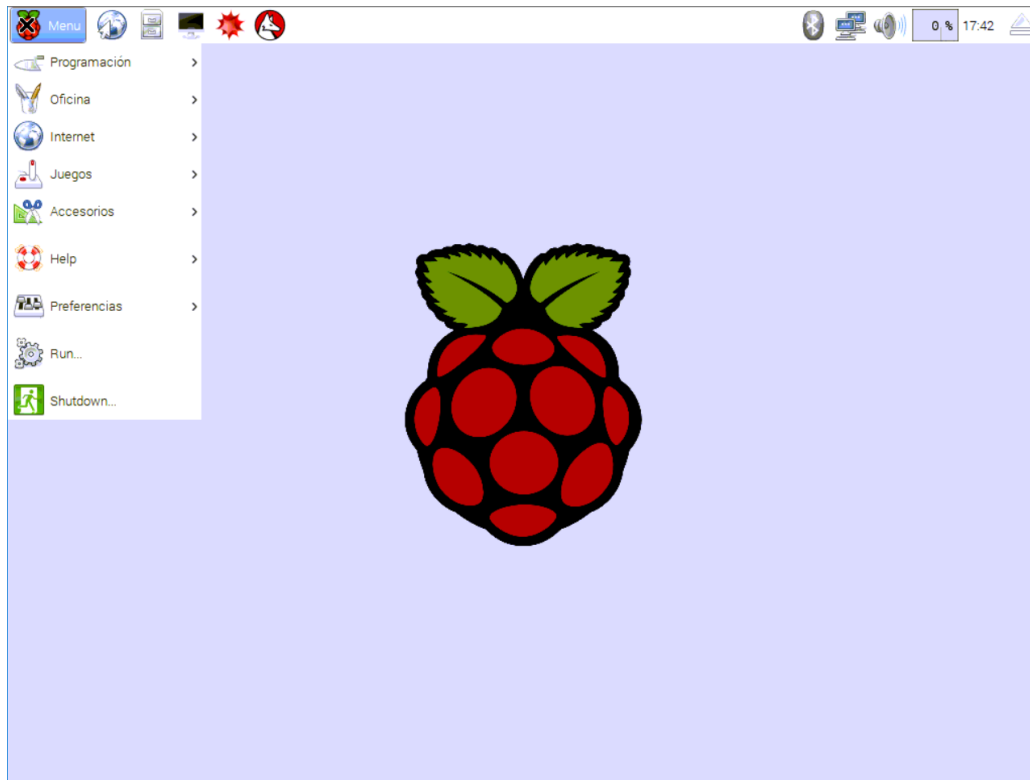


Imagen 2.16 Raspbian instalado

Fuente: (Raspbian, 2017)

2.1.2 CONFIGURACIÓN DE MIT APP INVENTOR 2

Para poder crear una aplicación mediante MIT App Inventor 2, es necesario tener creado una cuenta de correo electrónico en Gmail.

En el buscador de Google escriba MIT App inventor 2, aparecerán las opciones de la imagen 2.17, de un clic en la segunda opción de búsqueda, esta acción direccionará a la página principal de App Inventor (Imagen 2.18).

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

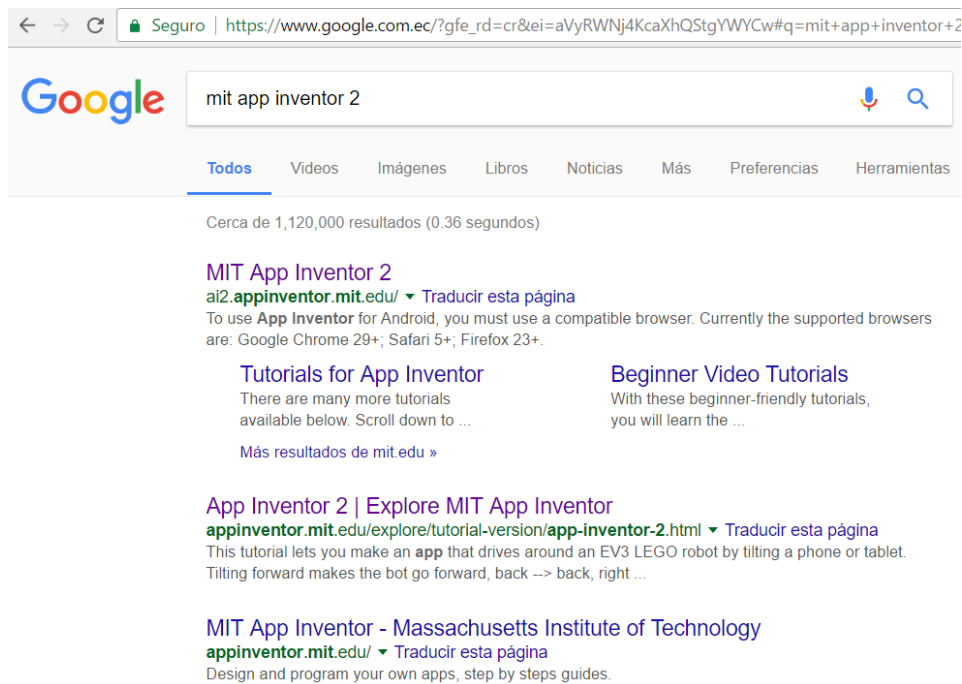


Imagen 2.17 Resultados de búsqueda de App Inventor

Fuente: (Google, 2017)

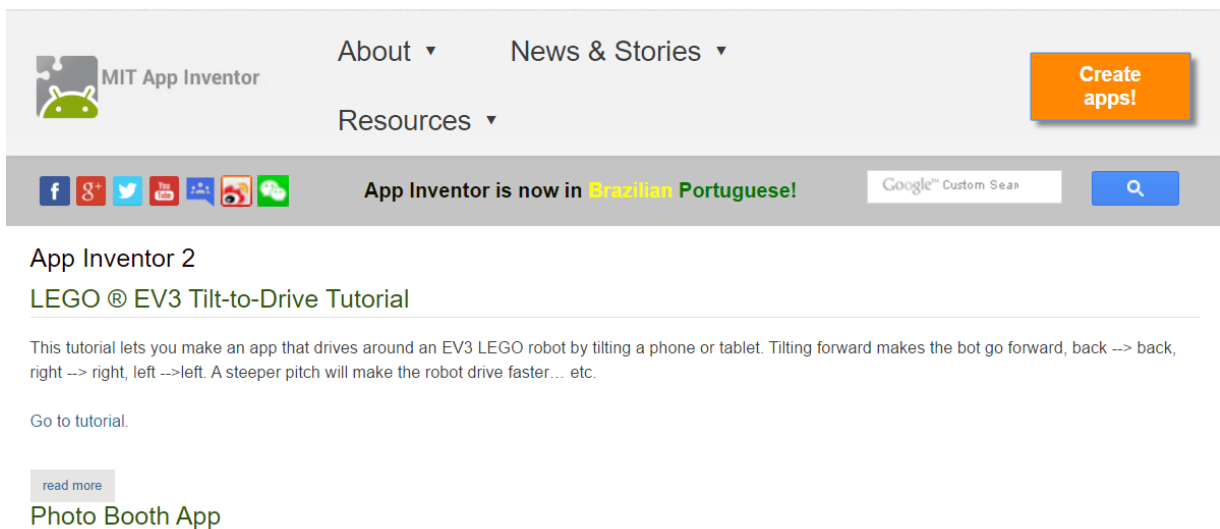


Imagen 2.18 Página principal de App Inventor

Fuente: (App inventor, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Para iniciar por primera vez una aplicación en App Inventor, pulse el botón de color naranja (Create apps!) ubicado en la parte superior derecha de la ventana, esto abrirá otra ventana en el navegador donde pide que ingrese el correo de Gmail. Ingrese el usuario y la contraseña del correo.

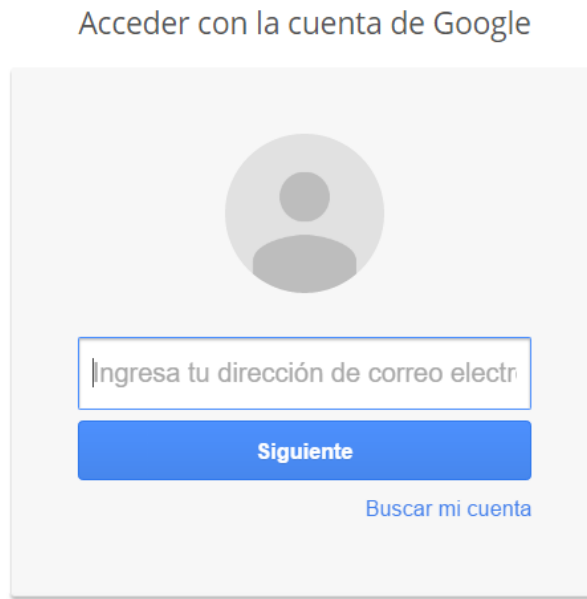


Imagen 2.19 Solicitud de correo para acceso a App Inventor

Fuente: (Google, 2017)

Cuando se ingresa el usuario y contraseña del correo de GMAIL, aparece la interface de programación de MIT App Inventor 2.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

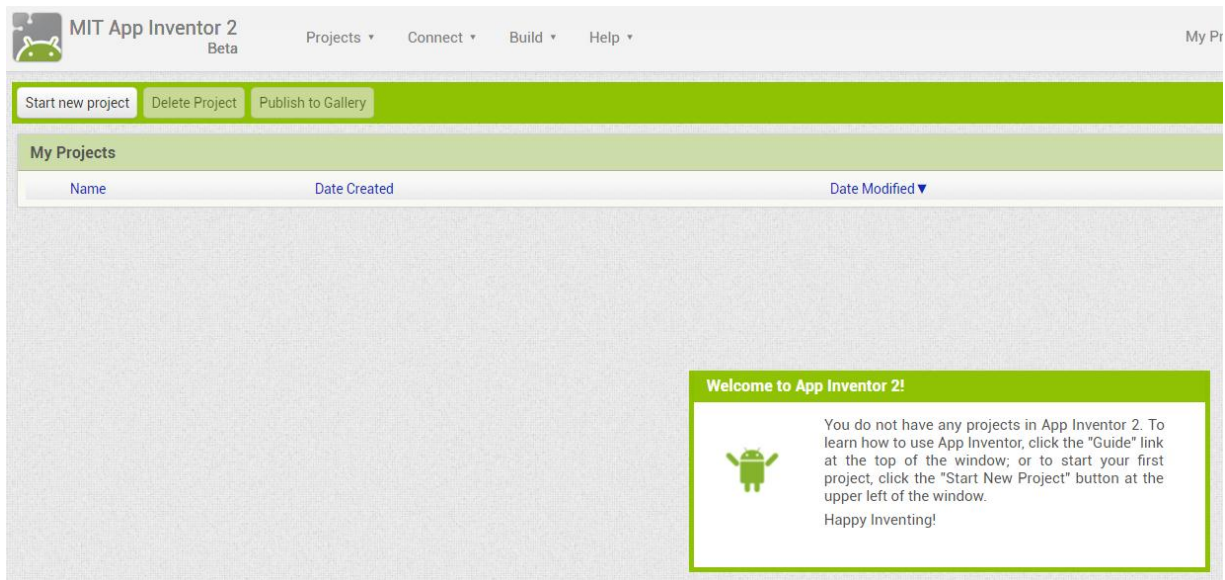


Imagen 2.20 Interface de programación de Mit App Inventor 2

Fuente: (App Inventor, 2017)

Es posible cambiar del idioma inglés al español, desde el menú de la misma interface.

Para no emplear el televisor, teclado y mouse que fueron necesarios para la instalación del software Raspbian y tener un acceso remoto al escritorio del sistema operativo instalado en el microcomputador Raspberry Pi mediante un software de Windows como es el caso de **TightVNC**, es necesario asignar una IP estática al Raspberry Pi.

2.1.3 ASIGNACIÓN DE IP ESTÁTICA AL RASPBERRY PI

Para asignar una dirección estática al Raspberry Pi se siguieron los siguientes pasos:

1. Conectar el Raspberry a un router que asigne direcciones dinámicas dentro de un rango.
2. Ingrese a la configuración del router y verifique la dirección dinámica que le fue asignado al Raspberry.
3. Para esta caso la dirección asignada por el router fue la 192.168.1.104
4. Una vez conocida la dirección Ip del Raspberry, hay que establecer comunicación con el mismo mediante el protocolo SSH y el software PuTTY.

Abra el software PuTTY, en la parte que esta etiquetado como Host Name, escriba la dirección IP del Raspberry Pi, en el casillero etiquetado como Port, escriba 22 que es el puerto que se

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

utilizará para comunicación SSH, en las opciones de Connection Type, seleccione SSH y pulse el botón Open.

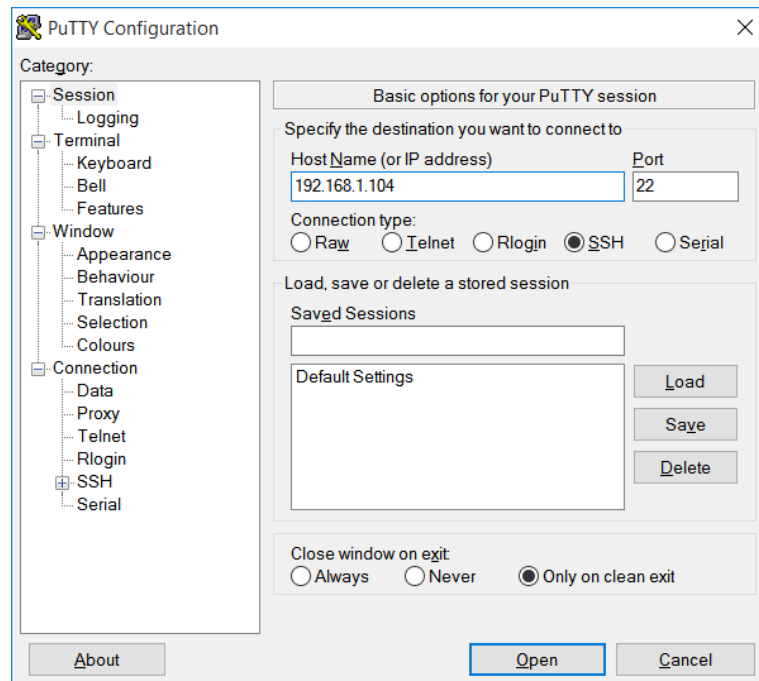


Imagen 2.21 Parámetros necesarios para la comunicación mediante SSH

Fuente: (PuTTY, 2017)

La primera vez que se establece la comunicación por SSH pedirá que confirme que vamos hacer la comunicación, para ello escribimos **yes** y le damos a enter. Después pide la **contraseña del usuario pi**, como no se cambió en el momento de la instalación se mantiene la que viene de fábrica que es **raspberry**, luego pulse enter y ya estamos dentro del microcomputador.

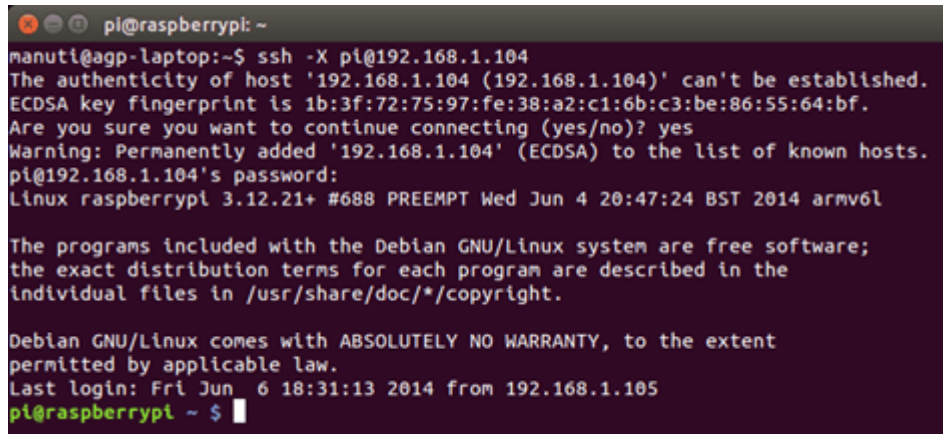


Imagen 2.22. Confirmación de la comunicación SSH

Fuente: (PuTTY, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

La imagen 2.24 muestra el Terminal para la comunicación entre el PuTTY y el Raspberry Pi mediante el protocolo SSH, luego del símbolo \$, se escriben los comandos Linux para configurar al microcomputador.



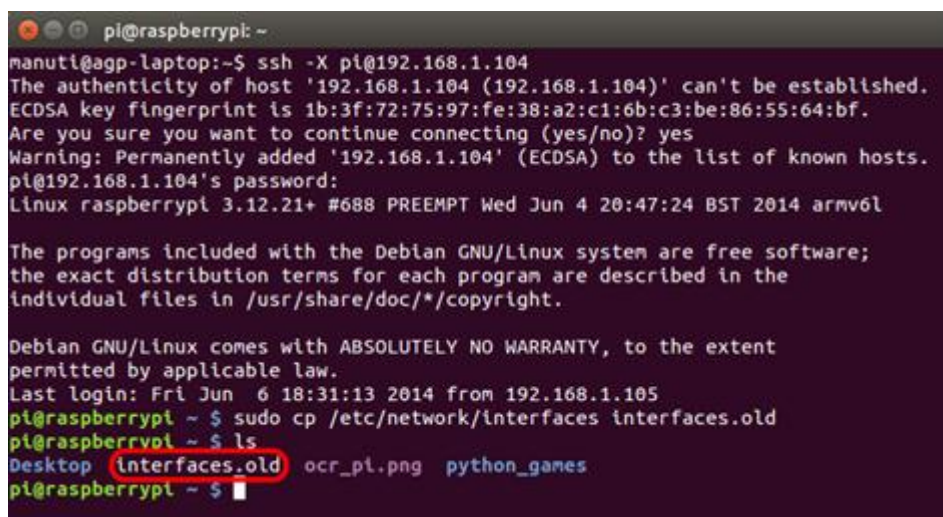
```
pi@raspberrypi: ~  
manuti@agp-laptop:~$ ssh -X pi@192.168.1.104  
The authenticity of host '192.168.1.104 (192.168.1.104)' can't be established.  
ECDSA key fingerprint is 1b:3f:72:75:97:fe:38:a2:c1:6b:c3:be:86:55:64:bf.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added '192.168.1.104' (ECDSA) to the list of known hosts.  
pi@192.168.1.104's password:  
Linux raspberrypi 3.12.21+ #688 PREEMPT Wed Jun 4 20:47:24 BST 2014 armv6l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Fri Jun 6 18:31:13 2014 from 192.168.1.105  
pi@raspberrypi ~ $
```

Imagen 2.23 Terminal para comunicación SSH

Fuente: (PuTTY, 2017)

Antes de modificar el archivo que contiene los parámetros de red, es necesario realizar una copia del mismo, para esto escriba lo siguiente:

sudo cp /etc/network/interfaces interfaces.old, pulse enter y luego escriba **ls**, pulse enter.



```
pi@raspberrypi: ~  
manuti@agp-laptop:~$ ssh -X pi@192.168.1.104  
The authenticity of host '192.168.1.104 (192.168.1.104)' can't be established.  
ECDSA key fingerprint is 1b:3f:72:75:97:fe:38:a2:c1:6b:c3:be:86:55:64:bf.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added '192.168.1.104' (ECDSA) to the list of known hosts.  
pi@192.168.1.104's password:  
Linux raspberrypi 3.12.21+ #688 PREEMPT Wed Jun 4 20:47:24 BST 2014 armv6l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Fri Jun 6 18:31:13 2014 from 192.168.1.105  
pi@raspberrypi ~ $ sudo cp /etc/network/interfaces interfaces.old  
pi@raspberrypi ~ $ ls  
Desktop interfaces.old ocr_pl.png python_games  
pi@raspberrypi ~ $
```

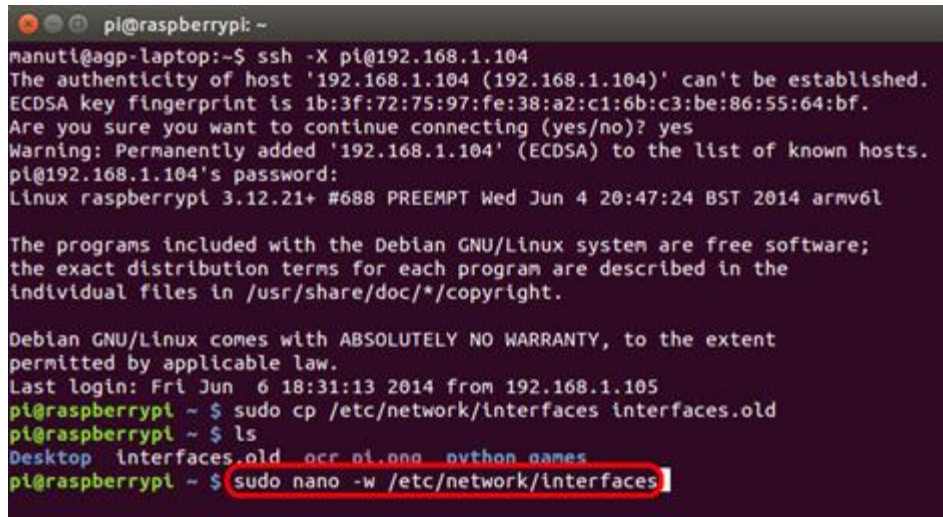
Imagen 2.24 Copia de respaldo del archive de red

Fuente: (PuTTY, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Para editar el archivo de configuración de los parámetros de red, escriba:

sudo nano -w /etc/network/interfaces, pulse enter



```
pi@raspberrypi: ~
manuti@agp-laptop:~$ ssh -X pi@192.168.1.104
The authenticity of host '192.168.1.104 (192.168.1.104)' can't be established.
ECDSA key fingerprint is 1b:3f:72:75:97:fe:38:a2:c1:6b:c3:be:86:55:64:bf.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.104' (ECDSA) to the list of known hosts.
pi@192.168.1.104's password:
Linux raspberrypi 3.12.21+ #688 PREEMPT Wed Jun 4 20:47:24 BST 2014 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jun 6 18:31:13 2014 from 192.168.1.105
pi@raspberrypi ~ $ sudo cp /etc/network/interfaces interfaces.old
pi@raspberrypi ~ $ ls
Desktop  interfaces.old  ocr  pi.png  python  games
pi@raspberrypi ~ $ sudo nano -w /etc/network/interfaces
```

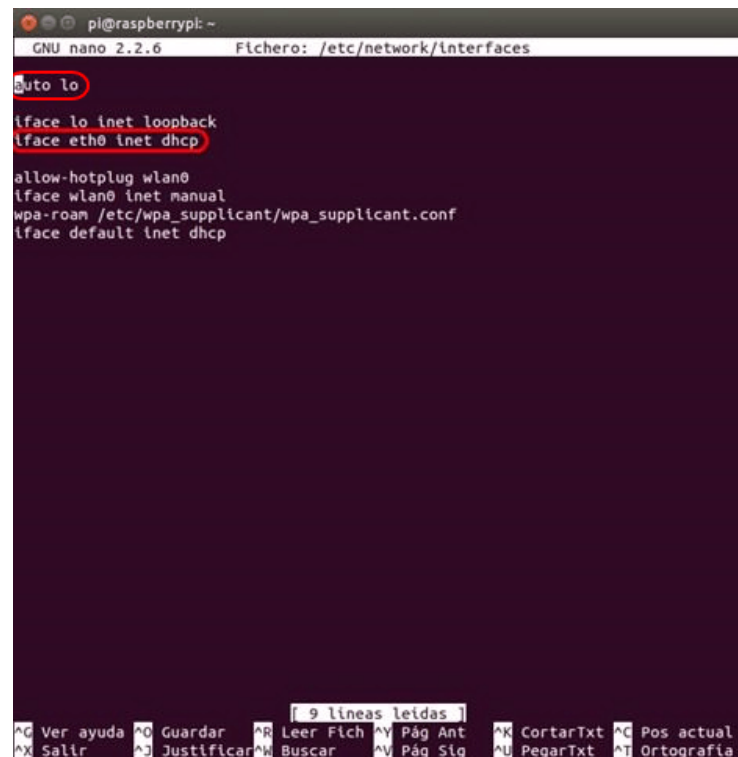
Imagen 2.25 Comando para editar archivo de red

Fuente: (PuTTY, 2017)

Se abre una aplicación que se llama **nano** y que es un editor muy básico pero más o menos sencillo de manejar. Con las teclas de las flechas, colóquese en la línea que dice:

iface eth0 inet dhcp o también puede ser **iface eth0 inet manual**

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



```
pi@raspberrypi: ~
GNU nano 2.2.6  Archivo: /etc/network/interfaces
Auto lo
iface lo inet loopback
iface eth0 inet dhcp

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp

9 líneas leídas
Ver ayuda  Guardar  Leer Fich  Pág Ant  CortarTxt  Pos actual
Salir  Justificar  Buscar  Pág Sig  PegarTxt  Ortografía
```

Imagen 2.26 Aplicación nano de Raspbian

Fuente: (PuTTY, 2017)

Borre la línea anterior y escriba lo siguiente:

iface eth0 inet static

address 192.168.0.25

netmask 255.255.255.0

gateway 192.168.0.1

Una vez cambiado, guarde los cambios y salga. Los comandos son:

- Para Guardar el archivo con los cambios pulsar la tecla **Ctrl** y al mismo tiempo la letra **o**
- Para salir del editor **nano** pulsar la tecla **Ctrl** y al mismo tiempo la letra **x**



Imagen 2.27 Comandos para guardar y salir de la aplicación nano

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Fuente: (PuTTY, 2017)

Aparece un mensaje en la parte inferior que dice si queremos **salvar el búfer modificado**, escriba una **S** y de un **enter**, luego indica el nombre del fichero, que debe ser el mismo, por lo que sin cambiarlo volvemos a pulsar **enter**.

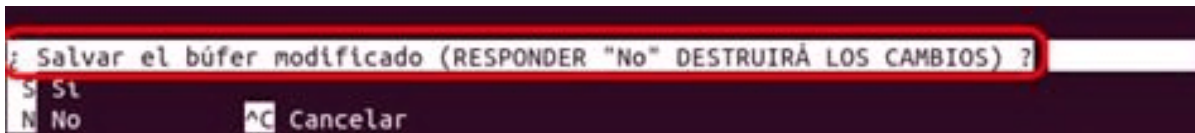


Imagen 2.28 Opción para modificar el archive de red

Fuente: (PuTTY, 2017)

Para que los cambios tengan efecto hay que reiniciar la conexión a la **Raspberry** con el siguiente comando:

sudo reboot

2.1.4 SOFTWARE DE CONTROL REMOTO

Para interactuar a través de Windows con el sistema operativo del Raspberry Pi, se utilizó TightVNC es un paquete gratuito de software de control remoto. Con TightVNC, se puede ver el escritorio del software Raspbian del Raspberry Pi, con el ratón y teclado locales, al igual que lo haría sentado en la parte delantera de ese equipo. TightVNC es:

- Libre , tanto para uso personal y comercial, con el código fuente completo,
- Útil en la administración, soporte técnico, la educación, y para muchos otros fines,
- Multiplataforma, disponible para Windows y Unix, con el cliente Java incluido,
- Compatible con el software VNC estándar, conforme a las especificaciones del protocolo RFB.

Para descargar este software ingrese al siguiente enlace:

<http://www.tightvnc.com/download.php>

Seleccione el tipo de sistema operativo que desee y luego descargue en su computador personal.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Una vez descargado el instalador, ejecute este archivo para proceder a la instalación, la primera ventana que se muestra es la de la imagen 2.30, pulse Next.

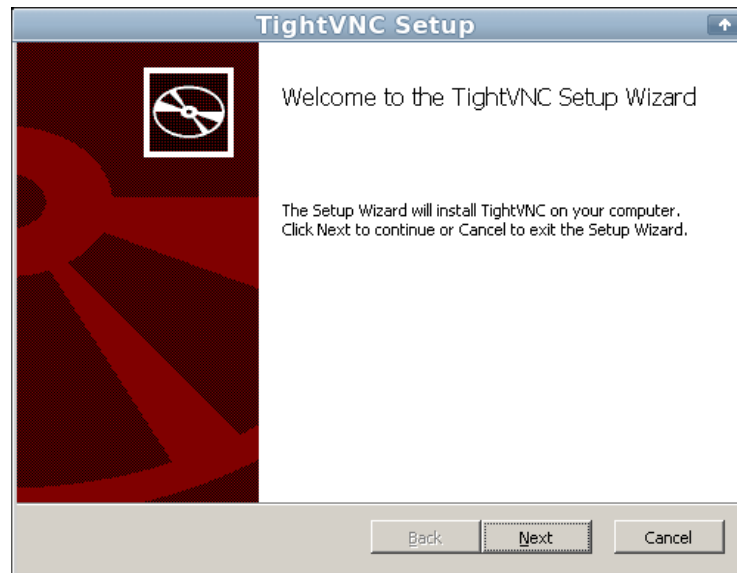


Imagen 2.29 Ventana de bienvenida a la instalación

Fuente: (TightVNC, 2017)

A continuación aparece la ventana de la licencia del software, marque *I accept the terms in the License Agreement*, luego pulse Next.

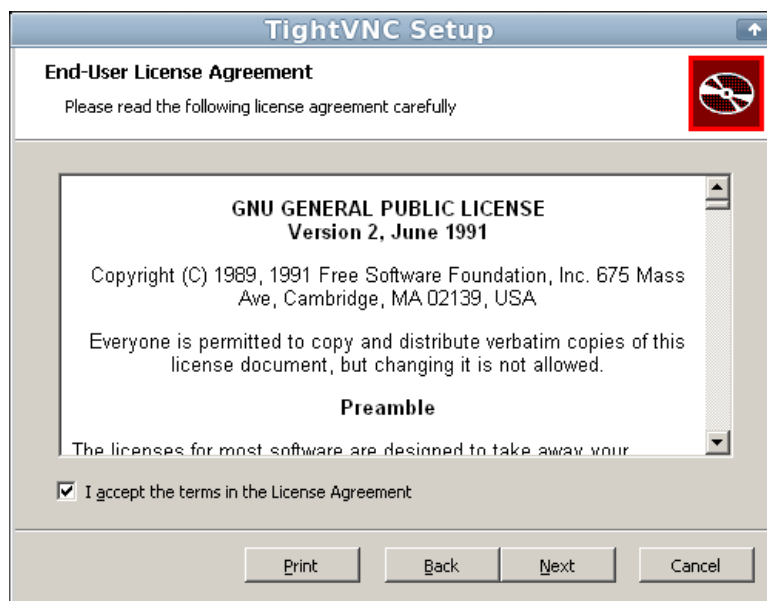


Imagen 2.30 Ventana de licencia GNU

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Fuente: (TightVNC, 2017)

La siguiente ventana que aparece es para escoger el tipo de instalación.

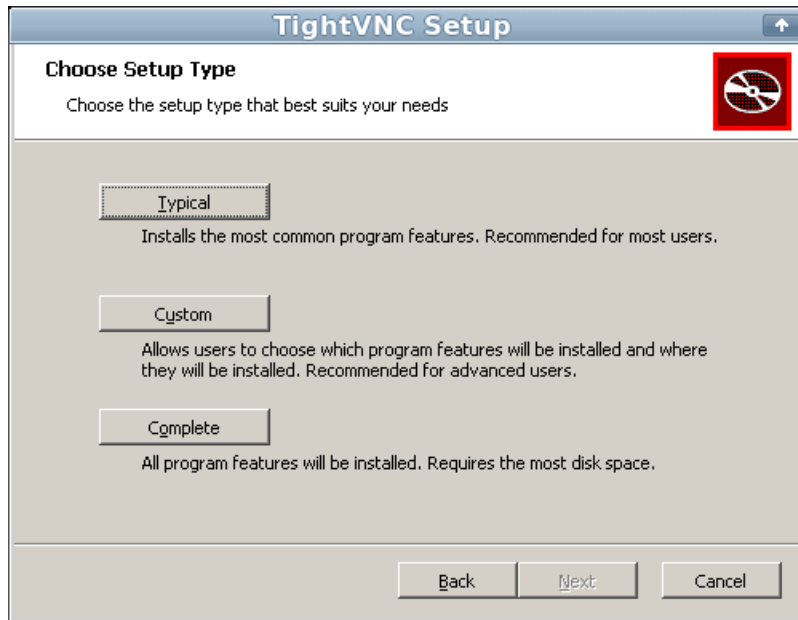


Imagen 2.31 Ventana de selección de tipo de instalación

Fuente: (TightVNC, 2017)

De un clic en Custom, para personalizar la instalación.

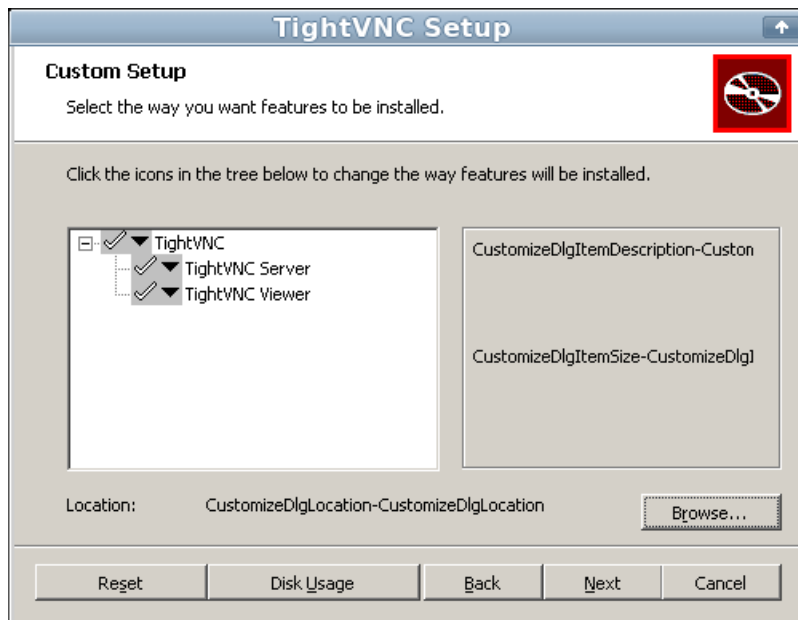


Imagen 2.32 Ventana de personalización de la instalación

Fuente: (TightVNC, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Dar click en el triángulo al lado de "TightVNC Server" y seleccionar la opción "**Don't Install**", el resultado de muestra en la imagen 2.34, dar clic en Next.

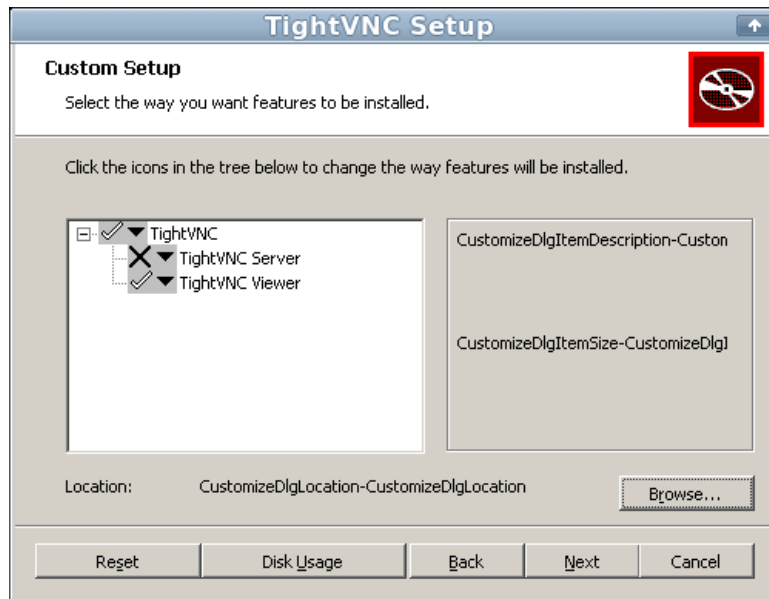


Imagen 2.33 No instalar Servidor TightVNC

Fuente: (TightVNC, 2017)

En la ventana de tareas adicionales, pulse Next.

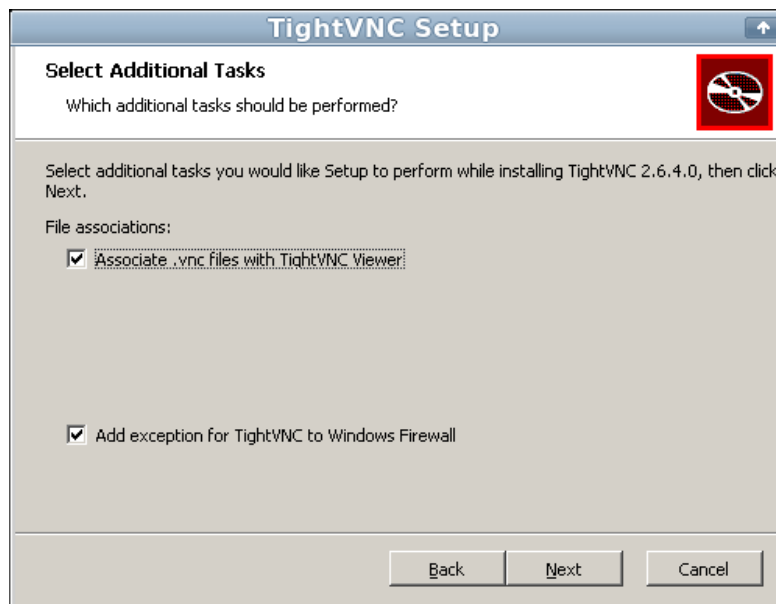


Imagen 2.34 Ventana de tareas adicionales

Fuente: (TightVNC, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Cuando aparece la ventana que indica el mensaje listo para la instalación de un clic en el botón Install.

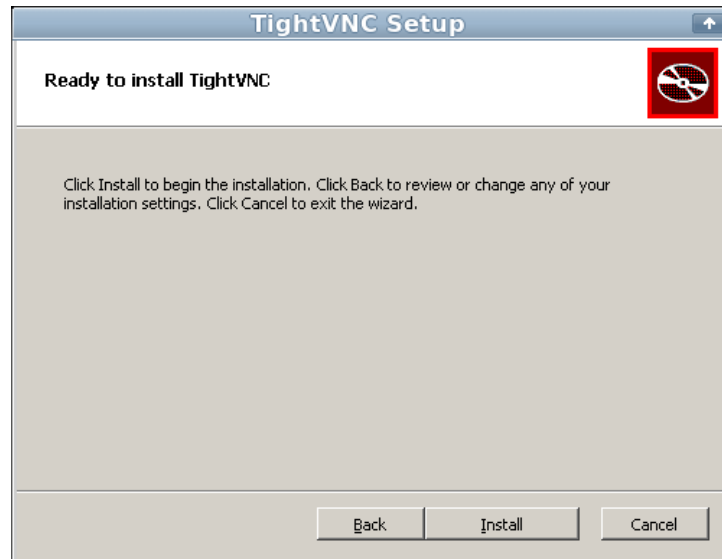


Imagen 2.35 Ventana de inicio de instalación

Fuente: (TightVNC, 2017)

Cuando finaliza el proceso de instalación aparece la ventana de la imagen 2.37, de un clic en el botón Finish.

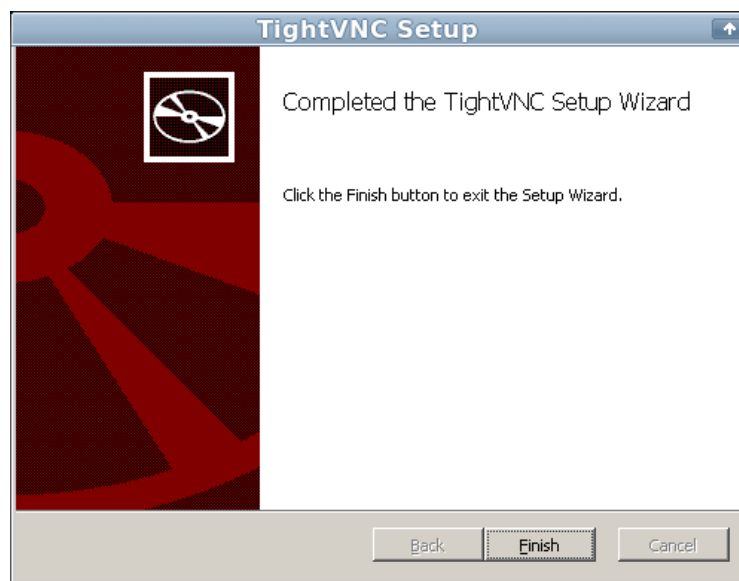


Imagen 2.36 Ventana que indica instalación completa

Fuente: (TightVNC, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

En el buscador de Windows escriba TightVNC Viewer, cuando aparezca de un clic sobre el nombre para abrir el programa, la interface de este programa se muestra en la imagen 3.28.

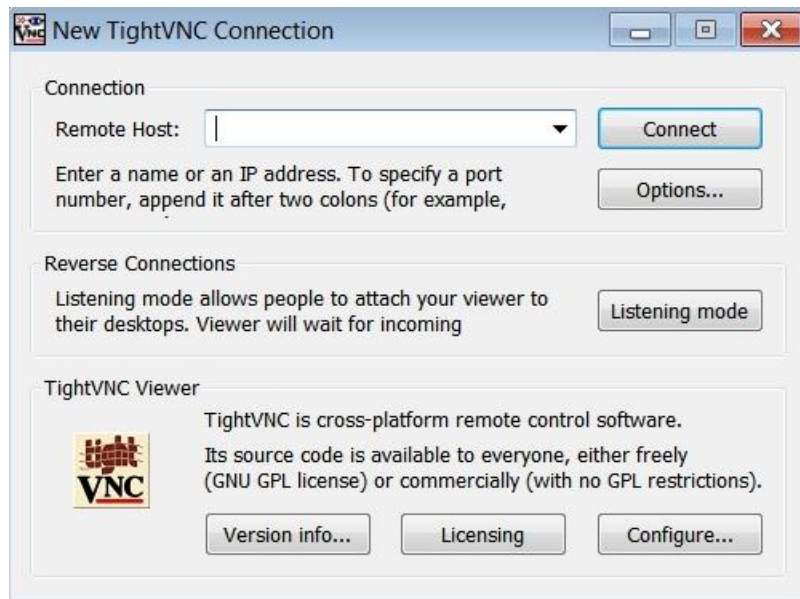


Imagen 2.37 Interface de TightVNC Viewer

Fuente: (TightVNC, 2017)

2.2 REQUERIMIENTO DE HARDWARE

2.2.1 CONEXIÓN DEL SENSOR DE TEMPERATURA

El sensor de temperatura empleado para este proyecto es el DS18B20, este dispositivo se polariza con la fuente de 3.3Vdc del Raspberry Pi, el terminal positivo de esta fuente está en el pin 1 del puerto GPIO del Raspberry Pi, el mismo que está etiquetado como 3V3, el terminal negativo corresponde al pin 9, etiquetado como Ground.

La salida del sensor, está conectada al pin 9, etiquetado como GPIO4, además es necesario conectar una resistencia de 4,7KΩ entre la salida del sensor y el terminal 3V3.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

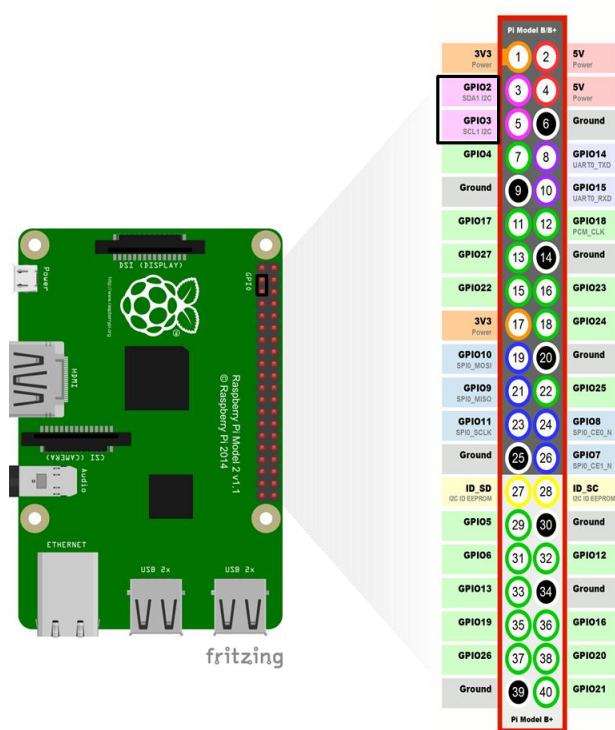


Imagen 2.38 Terminales del GPIO del Raspberry Pi

Fuente: (Jacho, 2017)

La conexión física realizada del sensor DS18B20 al puerto GPIO del Raspberry Pi se muestra en la imagen 2.40.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

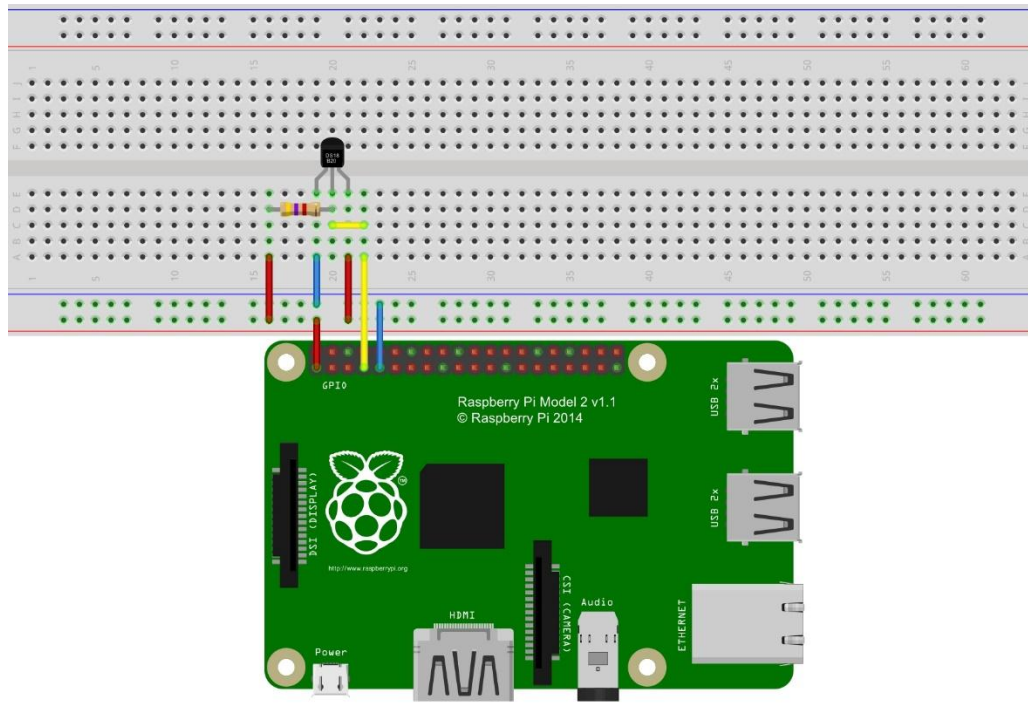


Imagen 2.39 Conexión del DS18B20 con el Raspberry Pi

Fuente: (Jacho, 2017)

2.2.2 CONEXIÓN DEL BLUETOOTH HC-06

El módulo Bluetooth HC-06, necesita de una fuente de alimentación de 5Vdc, el terminal de polarización positivo (VCC) se conecta al pin 2 del puerto GPIO del Raspberry Pi etiquetado como 5V, el terminal negativo (GND) se conecta al pin 20, etiquetado como Ground.

El terminal RXD del módulo Bluetooth HC-06, se conecta al pin 8 del puerto GPIO del Raspberry Pi, etiquetado como GPIO14.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

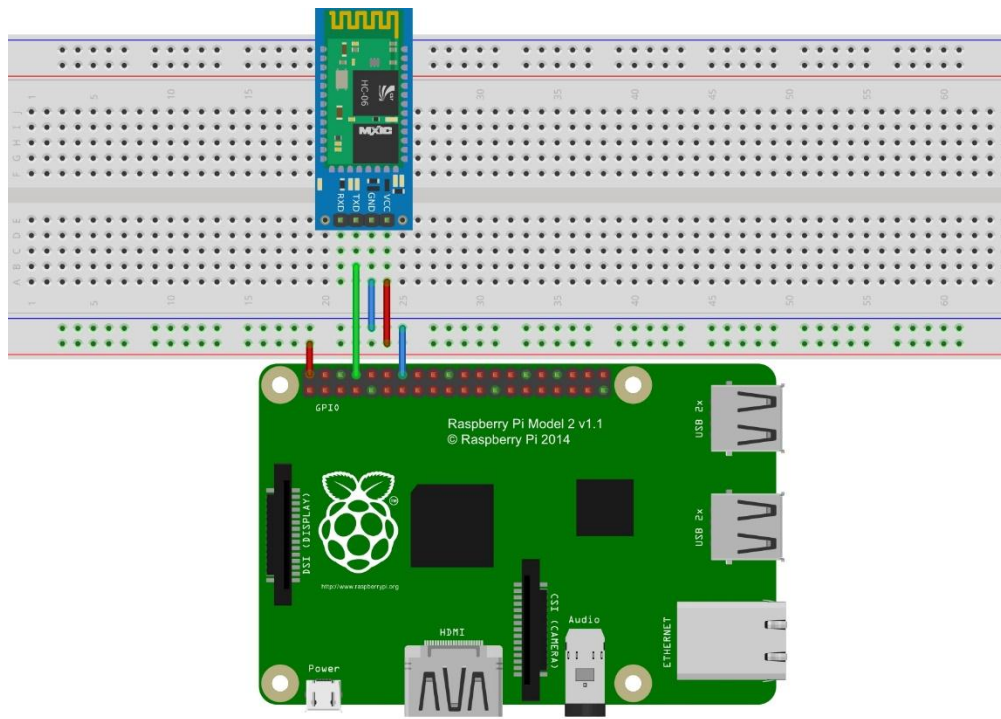


Imagen 2.40 Conexión del HC-06 al Rasperry Pi

Fuente: (Jacho, 2017)

2.3 METODOLOGÍA EN EL DESARROLLO DE APLICACIONES MÓVILES

2.3.1 METODOLOGÍAS ÁGILES

Las metodologías en general se clasifican según su enfoque y características esenciales, las más recientes, que se fueron gestando a finales del siglo pasado y que se han comenzado a manifestar desde principios del actual, se han denominado “metodologías ágiles” y surgen como una alternativa a las tradicionales, estas metodologías se derivan de la lista de los principios que se encuentran en el “Manifiesto Ágil”, y están basados en un desarrollo iterativo que se centra más en capturar mejor los requisitos cambiantes y la gestión de los riesgos, rompiendo el proyecto en iteraciones de diferente longitud, cada una de ellas generando un producto completo y entregable; e incremental donde un producto se construye bloque a bloque durante todo el ciclo de vida de desarrollo del producto, las iteraciones individuales deben producir alguna característica completamente funcional o mejorada, su principal objetivo es reducir el tiempo

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

de desarrollo, del mismo modo que con el modelo en cascada o waterfall que fuera introducido por Royce en 1970 y usado inicialmente para desarrollo de software, pero expandido por Boehm en 1981, donde todos los requisitos se analizan antes de empezar a desarrollar, sin embargo, los requisitos se dividen en “incrementos” independientemente funcionales (Amaya Balaguera, 2013).

La aparición de las metodologías ágiles no puede ser asociada a una única causa, sino a todo un conjunto de ellas, si bien es cierto que la mayoría de autores lo relacionan con una reacción a las metodologías tradicionales, ¿cuáles fueron las causas de esta reacción?, los factores que comúnmente se mencionan son la pesadez, lentitud de reacción y exceso de documentación, en definitiva, falta de agilidad de los modelos de desarrollo formales; otro punto importante sería la explosión de la red, las aplicaciones Web y las aplicaciones móviles, así como el crecimiento notorio del movimiento open source (Amaya Balaguera, 2013).

A todo esto, se puede añadir un cambio bastante importante, en cuanto a la demanda del mercado del software, cada vez más orientada a la Web y a dispositivos móviles, con requisitos muy volátiles y en constante cambio, que requieren tiempos de desarrollo cada vez más cortos, lo que provocó que las empresas se fijaran más en nuevos desarrolladores, con nuevos métodos “amateurs” que se combinan con técnicas de las metodologías formales. Los modelos de desarrollo de las comunidades open source pudieron ciertamente determinar la aparición de las metodologías ágiles, pero cada autor determina el surgimiento de las metodologías ágiles de diferentes maneras (Amaya Balaguera, 2013).

2.3.2 METODOLOGÍA MOBILE DEVELOPMENT PROCESS SPIRAL

Esta propuesta metodológica utiliza el modelo de desarrollo en espiral como base, e incorpora procesos de evaluación de la usabilidad, priorizando la participación del usuario en todos los procesos del ciclo de vida de diseño, con el fin de garantizar un diseño centrado en el usuario, aun cuando se trata de un modelo de proceso orientado a proyectos grandes y costosos, ya que está destinado a ser un modelo de reducción de riesgos (Amaya Balaguera, 2013).

El proceso permite a los desarrolladores de aplicaciones móviles, detallar los criterios de usabilidad de la aplicación, el primer paso es identificar a los usuarios, las tareas y los contextos en los que se utilizará la aplicación móvil, el siguiente paso es dar prioridad a los atributos de usabilidad, identificar qué atributos son los más importantes para la aplicación, y para cada uno

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

definir un conjunto de métricas para verificar el grado en que se cumplen en la aplicación final (Amaya Balaguera, 2013).

El proceso de desarrollo de aplicaciones móviles en espiral contempla cinco (5) iteraciones, para cada una de ellas tres (3) tareas (determinación de requisitos, diseño y prueba) y finaliza cada iteración con la planificación de la siguiente; en la primera iteración se determinan los requisitos del sistema y se identifican usuarios, tareas y contextos en los que se utilizará la aplicación. Luego, se definen y priorizan los atributos de facilidad de uso y se identifican métricas para cada atributo; se dibuja un prototipo de la interfaz de aplicación y se realiza la prueba del prototipo, los desarrolladores podrán utilizar diferentes técnicas de usabilidad para medir el valor de cada atributo (Amaya Balaguera, 2013).

En la segunda iteración el equipo de desarrollo recogerá más datos y requisitos, explorará si hay más usuarios potenciales, tareas y contextos en los que se utilizará la aplicación. A continuación, los atributos de usabilidad se redefinen y son priorizados, como resultado, los desarrolladores alterarán las métricas para acomodar los requisitos añadidos; en el diseño se realiza un prototipo de alta fidelidad de la interfaz y se realizan las pruebas, utilizando técnicas de usabilidad para cada atributo, la calificación se compara con los resultados de la iteración anterior (Amaya Balaguera, 2013).

En la tercera iteración los desarrolladores pueden identificar y priorizar los atributos de usabilidad con mayor claridad utilizando los resultados de la iteración anterior; se desarrolla el diseño de todo el sistema y se realiza la versión alfa con sus respectivas pruebas, el equipo de desarrollo compara los resultados con la calificación de la iteración anterior (Amaya Balaguera, 2013).

En la cuarta iteración los resultados de la iteración anterior son utilizados para identificar y dar prioridad a los atributos de facilidad de uso; se desarrolla la versión beta y se libera para su evaluación por parte del cliente (Amaya Balaguera, 2013).

En la quinta iteración se desarrolla el producto final; se realiza una evaluación de facilidad de uso, la calificación de cada atributo se calcula y se compara con la calificación de la fase anterior. Una alteración en el producto final se realiza sobre la base de los resultados y se libera al producto (Amaya Balaguera, 2013).

DESARROLLO E IMPLEMENTACIÓN DE LA APLICACIÓN

El proyecto consiste en monitorear la temperatura ambiente mediante la integración de un microcomputador Raspberry Pi y una aplicación móvil.

Raspberry Pi es el encargado de adquirir el valor de temperatura con la ayuda de un sensor digital DS18B20, este se comunica con el microcomputador a través de la interface One Wire. El valor de temperatura se envía a la aplicación móvil mediante bluetooth.

3.1 PROGRAMACIÓN EN PYTHON PARA LA ADQUISICIÓN DE LA TEMPERATURA

3.1.1 ACCESO REMOTO AL RASPBERRY PI DESDE WINDOWS

Para el acceso remoto al escritorio de Raspbian, abra TightVNC Viewer y escriba la dirección IP estática asignada al Raspberry, seguido de los dos puntos y el número, esto especifica que es el dispositivo número uno, 192.168.0.25:1. La imagen 3.1 muestra lo explicado.

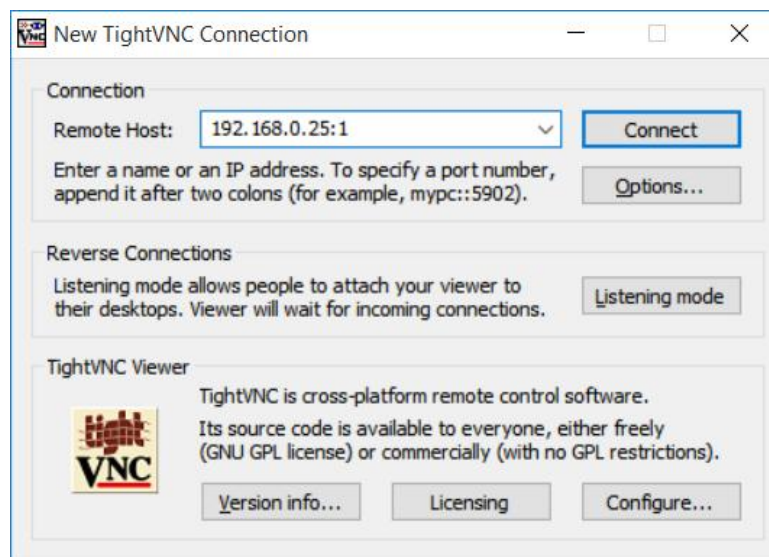


Imagen 3.1 Dirección IP del equipo remoto

Fuente: (TightVNC, 2017)

Una vez escrita la dirección IP del equipo remoto, pulse en el botón Connect, aparece una ventana autenticación donde pide la contraseña de acceso al dispositivo, escriba *raspberry* en la casilla Password y pulse el botón OK.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

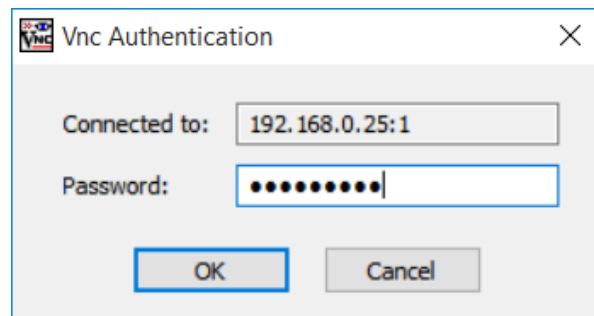


Imagen 3.2 Ventana de Autenticación de acceso

Fuente: (TightVNC, 2017)

En la imagen 3.3 se observa el escritorio de Raspbian, cuyo acceso es remoto desde el escritorio de Windows con la ayuda de TightVNC Viewer; es decir, se puede utilizar el teclado y mouse del computador que contiene el sistema operativo Windows.

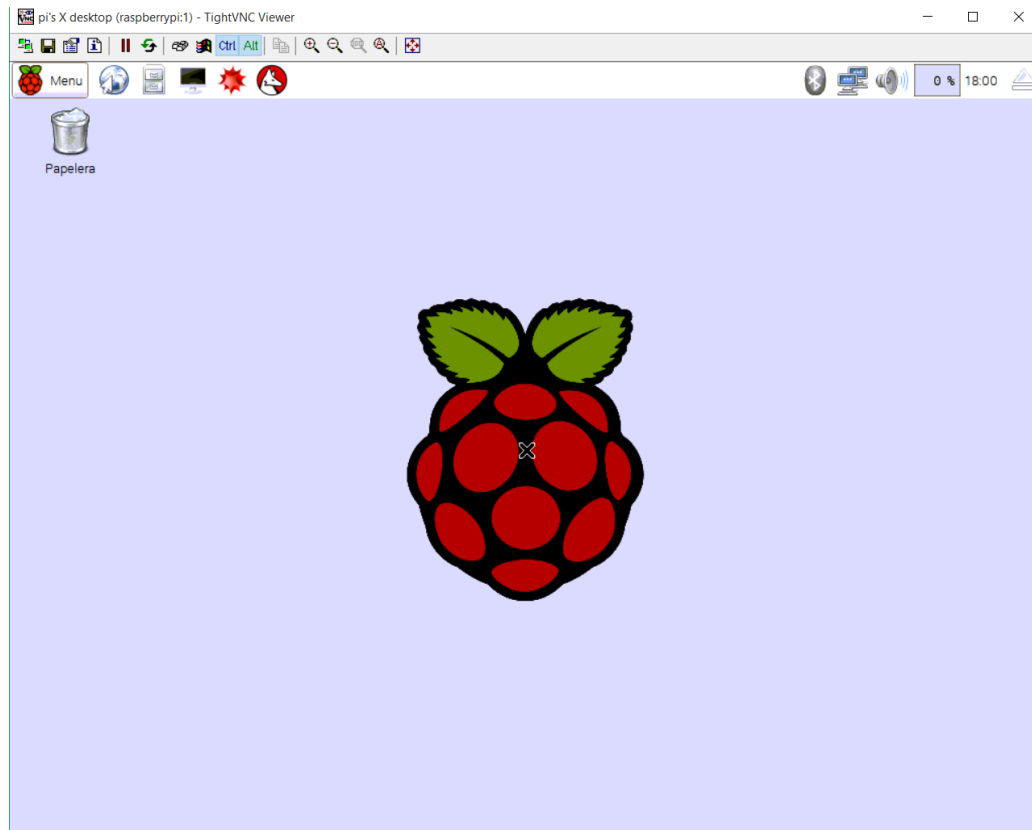


Imagen 3.3 Escritorio remote de Raspbian

Fuente: (TightVNC, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

3.1.2 EXPLICACIÓN DE LA PROGRAMACIÓN EN PYTHON PARA LA ADQUISICIÓN DE TEMPERATURA AMBIENTE

Desde el menú de Raspbian, en la opción programación abra Python 2, este procedimiento se observa en la imagen 3.4.

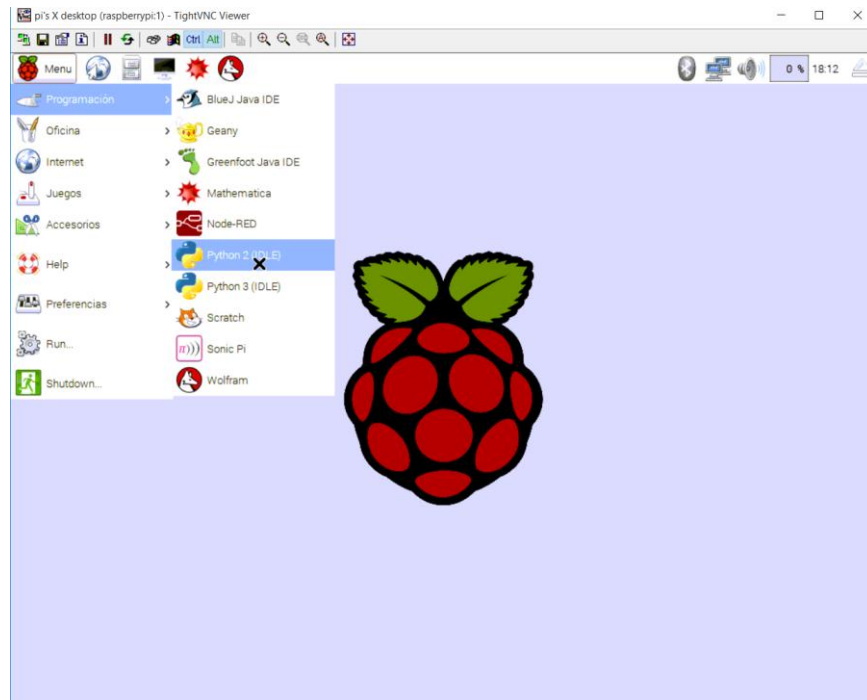


Imagen 3.4 Procedimiento para abrir Python 2

Fuente: (TightVNC, 2017)

El entorno de desarrollo integrado (IDLE) de Python se observa en la imagen 3.5, esta es la ventana donde muestran los resultados de la programación.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

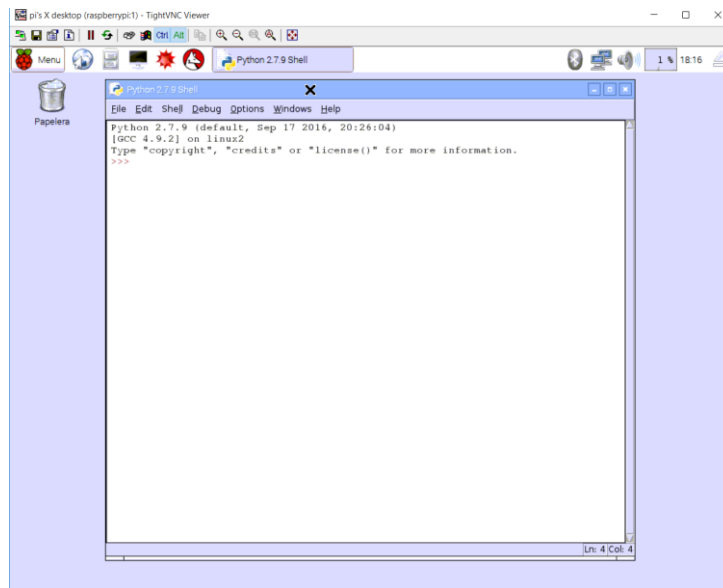


Imagen 3.5 IDLE de Python

Fuente: (TightVNC, 2017)

Para crear un nuevo programa, desde el menú File, seleccione New File o pulse la combinación de teclas Ctrl+N, la imagen 3.6 muestra lo explicado.

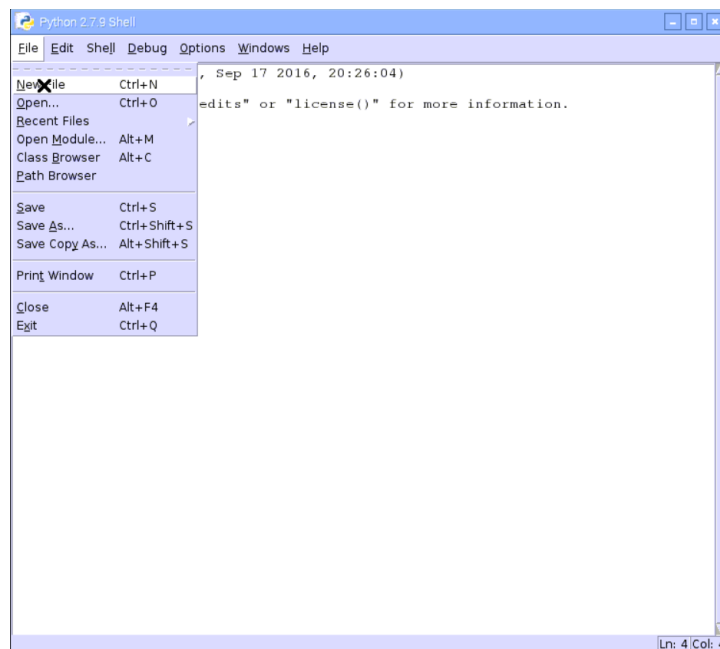


Imagen 3.6 Creación de Nuevo programa

Fuente: (Python, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Se abrirá otra ventana, asigne un nombre para el programa.

Antes de iniciar con la programación es necesario agregar al final del archivo Config.txt, una línea de comando que indique que se va a utilizar la interface One Wire. Para lo cual abra el terminal de comando del Raspbian, escriba lo siguiente:

sudo nano /boot/config.txt

Este comando abrirá en modo administrador el archivo config.txt para modificarlo. En la imagen 3.7 se observa la línea de comando escrito en el terminal.

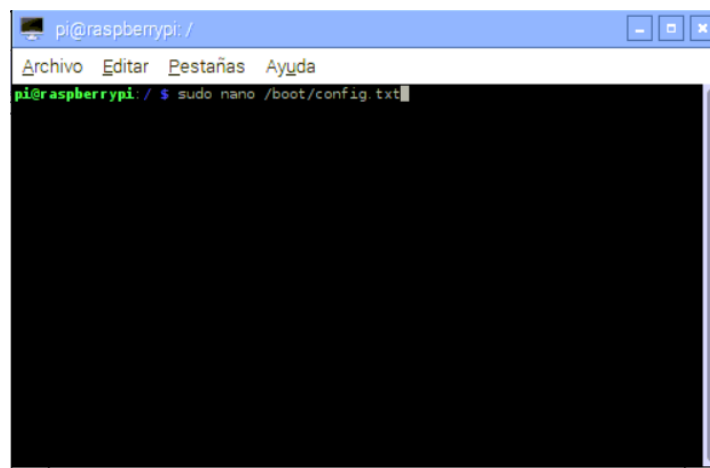


Imagen 3.7 Línea de comando para abrir el fichero config

Fuente: (Raspbian, 2017)

Al pulsar una vez escrito la línea de comando de la imagen 3.7, aparece la ventana donde se muestra en archivo config.txt, con la ayuda de la tecla flecha hacia abajo, ubíquese al final del archivo y escriba lo siguiente:

dtoverlay=w1-gpio

Guarde los cambios con la combinación de teclas Ctrl+O, luego salga del archivo mediante la combinación de teclas Ctrl+X.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Una vez en el terminal para que los cambios surtan efecto, reinicie el Raspberry Pi escribiendo *sudo reboot* en el terminal de comandos.

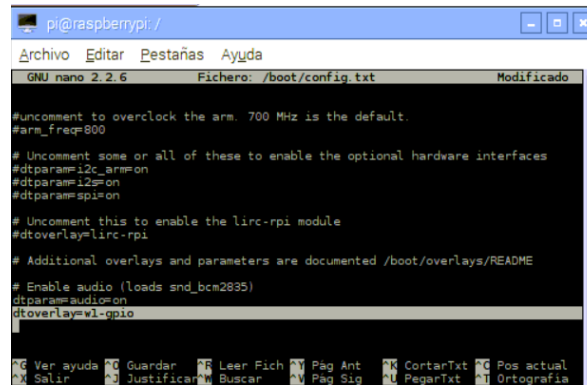


Imagen 3.8 Modificación del fichero config.txt

Fuente: (Raspbian, 2017)

Todo programa hecho en Python, inicia con la declaración *import* que encuentra un módulo o librería y ejecuta si es necesario. Las librerías utilizadas para este ejemplo son:

- import os
- import glob
- import time

La primera librería *import os*, proporciona múltiples funciones para interactuar con el sistema operativo.

La librería *import glob*, proporciona una función para crear listas de archivos a partir de búsquedas en directorios.

La librería *import time*, proporciona una función para crear retardos de tiempo.

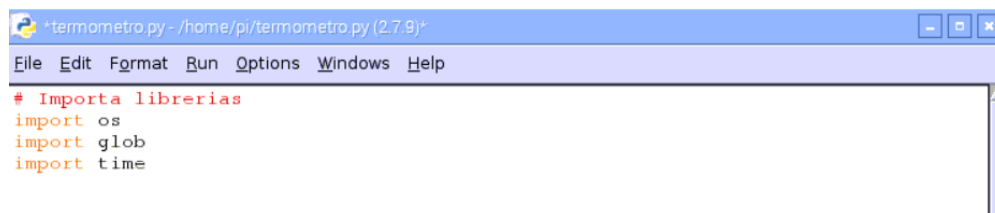


Imagen 3.9 Librerías necesarias para la adquisición de temperatura

Fuente: (Python, 2017)

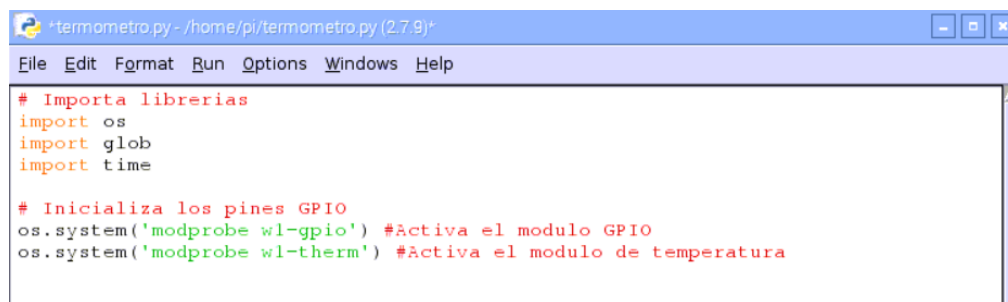
“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

La imagen 3.9 muestra las líneas de comando para importar las librerías necesarias para el programa de adquisición de temperatura. Los comentarios inician con el símbolo (#) y las declaraciones son de color naranja.

Luego es necesario activar el GPIO para adquisición de temperatura mediante la interface One Wire, para esto se escribe:

```
os.system('modprobe w1-gpio')  
os.system('modprobe w1-therm')
```

La imagen 3.10 muestra cómo queda en el IDLE de Python



```
*termometro.py - /home/pi/termometro.py (2.7.9)*  
File Edit Format Run Options Windows Help  
# Importa librerías  
import os  
import glob  
import time  
# Inicializa los pines GPIO  
os.system('modprobe w1-gpio') #Activa el modulo GPIO  
os.system('modprobe w1-therm') #Activa el modulo de temperatura
```

Imagen 3.10 Inicialización de los pines GPIO

Fuente: (Python, 2017)

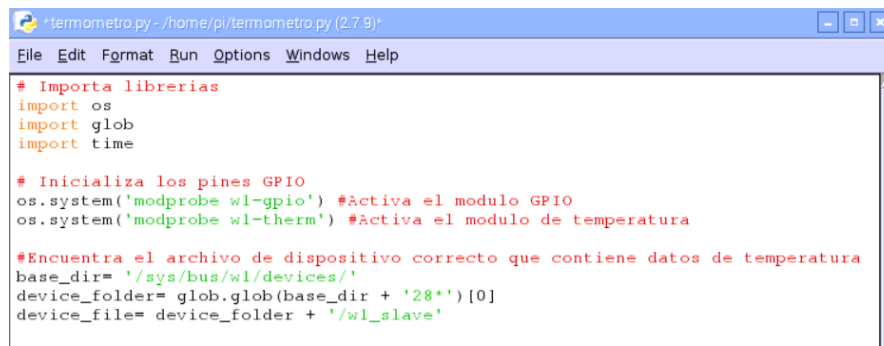
El sensor debe conectarse en el pin GPIO 4.

Una vez que se inicializa los pines del GPIO, es necesario buscar un directorio que inicia con el número 28 seguido de un número largo, éste corresponde al sensor de temperatura, a continuación, se debe concatenar con *w1_slave*, para formar un archivo que será llamado por una función de lectura que se creara. Las líneas de comando para este son:

```
base_dir= '/sys/bus/w1/devices/'  
device_folder= glob.glob(base_dir + '28*')[0]  
device_file= device_folder + '/w1_slave'
```

En la imagen 3.11 se muestra las líneas de comando escritas para este propósito.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



```
*termometro.py - /home/pi/termometro.py (2.7.9)*
File Edit Format Run Options Windows Help

# Importa librerias
import os
import glob
import time

# Inicializa los pines GPIO
os.system('modprobe wl-gpio') #Activa el modulo GPIO
os.system('modprobe wl-therm') #Activa el modulo de temperatura

#Encuentra el archivo de dispositivo correcto que contiene datos de temperatura
base_dir= '/sys/bus/wl/devices/'
device_folder= glob.glob(base_dir + '28*')[0]
device_file= device_folder + '/wl_slave'
```

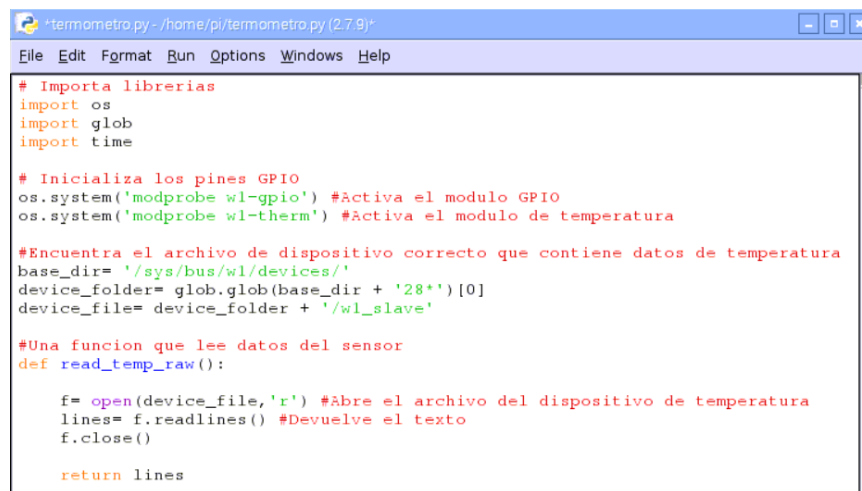
Imagen 3.11 Búsqueda del directorio del sensor

Fuente: (Python, 2017)

Se creó una función para leer el archivo de la trama completa que envía el sensor una vez que se buscó y se guardó la ruta de la carpeta. La función es:

```
def read_temp_raw():
f= open(device_file,'r') #Abre el archivo del dispositivo de temperatura
lines= f.readlines() #Devuelve el texto
f.close()
return lines
```

La imagen 3.12, muestra la función escrita en Python para leer la trama enviada por el sensor de temperatura DS18B20.



```
*termometro.py - /home/pi/termometro.py (2.7.9)*
File Edit Format Run Options Windows Help

# Importa librerias
import os
import glob
import time

# Inicializa los pines GPIO
os.system('modprobe wl-gpio') #Activa el modulo GPIO
os.system('modprobe wl-therm') #Activa el modulo de temperatura

#Encuentra el archivo de dispositivo correcto que contiene datos de temperatura
base_dir= '/sys/bus/wl/devices/'
device_folder= glob.glob(base_dir + '28*')[0]
device_file= device_folder + '/wl_slave'

#Una funcion que lee datos del sensor
def read_temp_raw():

    f= open(device_file,'r') #Abre el archivo del dispositivo de temperatura
    lines= f.readlines() #Devuelve el texto
    f.close()

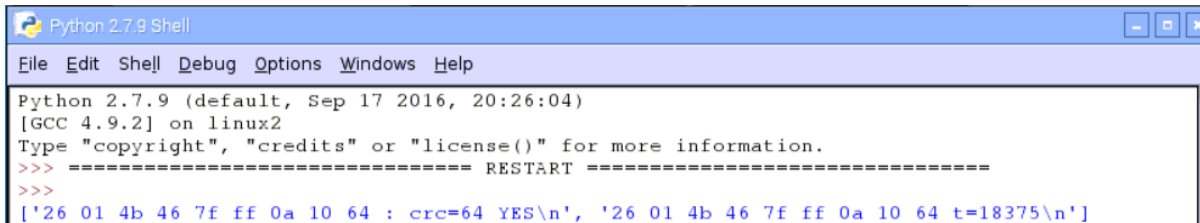
    return lines
```

Imagen 3.12 Función para leer trama enviada por DS18B20

Fuente: (Python, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

La función retorna dos líneas, esto se muestra en la imagen 3.13.



```
Python 2.7.9 Shell
File Edit Shell Debug Options Windows Help
Python 2.7.9 (default, Sep 17 2016, 20:26:04)
[GCC 4.9.2] on linux2
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
['26 01 4b 46 7f ff 0a 10 64 : crc=64 YES\n', '26 01 4b 46 7f ff 0a 10 64 t=18375\n']
```

Imagen 3.13 Trama de datos enviada por el DS18B20

Fuente: (Python, 2017)

El carácter ('\n'), representa el salto de línea, al final de la primera línea el sensor responde con YES cuando envía un dato correcto y al final de la segunda línea envía el valor de la temperatura precedido de (t=).

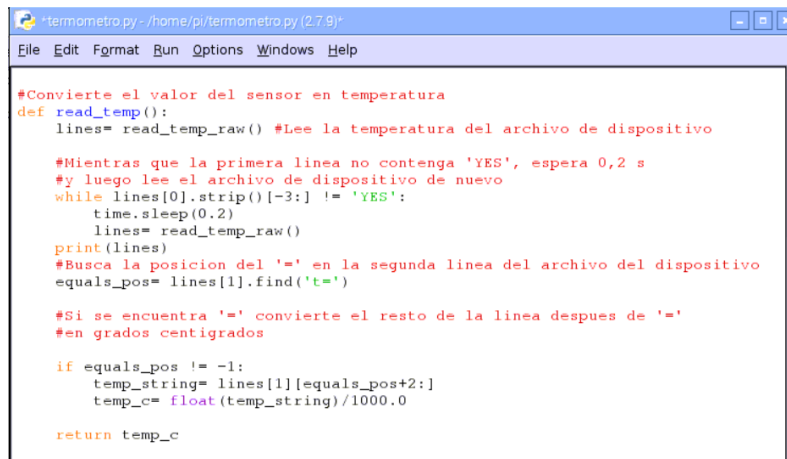
Una función para convertir la trama recibida en un valor de temperatura en grados centígrados y en formato entero es la siguiente:

```
def read_temp():
    lines= read_temp_raw()
    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines= read_temp_raw()
    equals_pos= lines[1].find('t=')
    if equals_pos != -1:
        temp_string= lines[1][equals_pos+2:]
        temp_c= float(temp_string)/1000.0
        temperatura=int(temp_c)
    return temperatura
```

Esta función llama a la que lee la trama enviada por el sensor, si no recibe YES, espera 200 ms y nuevamente vuelve a llamar a la función, cuando ya recibe YES, busca en la segunda línea la cadena “t=” porque a partir de la letra t, recibe un valor, para obtener el valor de temperatura

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

real, es necesario desplazar dos posiciones a la derecha para discriminar la cadena “t=”, es valor como esta en tipo de dato string, es necesario convertirle a flotante para proceder a dividir para mil, ese valor flotante se convierte a entero y se guarda en la variable temperatura. La escritura de esta función se observa en la imagen 3.14.



```
*termometro.py - /home/pi/termometro.py (2.7.9)*
File Edit Format Run Options Windows Help

#Convierte el valor del sensor en temperatura
def read_temp():
    lines= read_temp_raw() #Lee la temperatura del archivo de dispositivo

    #Mientras que la primera linea no contenga 'YES', espera 0,2 s
    #y luego lee el archivo de dispositivo de nuevo
    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines= read_temp_raw()
    print(lines)
    #Busca la posicion del '=' en la segunda linea del archivo del dispositivo
    equals_pos= lines[1].find('t=')

    #Si se encuentra '=' convierte el resto de la linea despues de '='
    #en grados centigrados

    if equals_pos != -1:
        temp_string= lines[1][equals_pos+2:]
        temp_c= float(temp_string)/1000.0

    return temp_c
```

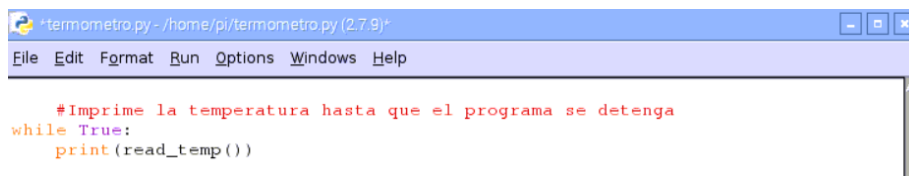
Imagen 3.14 Función read_temp()

Fuente: (Python, 2017)

Una vez realizada la configuración los pines del GPIO, asignada la ruta donde se encuentra la carpeta del sensor y creadas las funciones para leer la trama y convertir a temperatura, mediante la estructura **while**, se ejecuta la instrucción **print** para llamar a la función **read_temp()** e imprimir el valor de la temperatura en la consola del IDLE de Python, de la siguiente manera:

print(read_temp())

La imagen 3.15 muestra cómo se debe escribir en Python.



```
*termometro.py - /home/pi/termometro.py (2.7.9)*
File Edit Format Run Options Windows Help

#Imprime la temperatura hasta que el programa se detenga
while True:
    print(read_temp())
```

Imagen 3.15 Función principal del programa en Python

Fuente: (Python, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Una vez escrito todo el programa en Python para comprobar su funcionamiento, guarde el archivo, desde el menú Run, seleccione Run Module o pulse la tecla F5, este proceso se indica en la imagen 3.16.

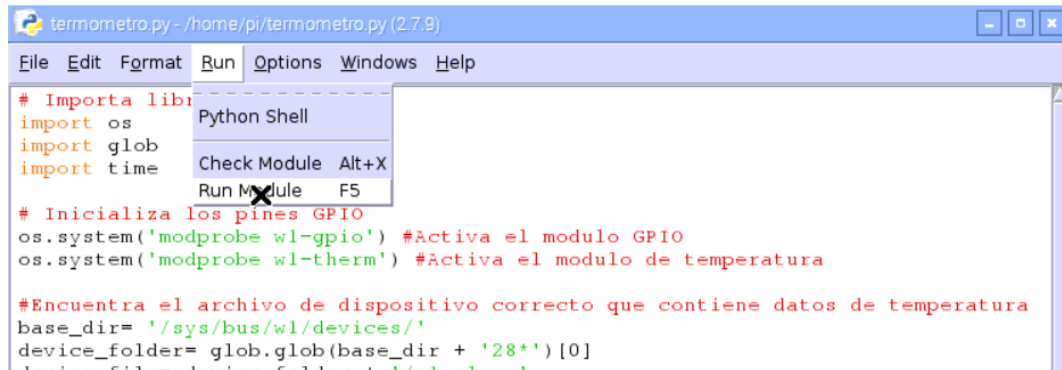


Imagen 3.16 Ejecución de programa en Python

Fuente: (Python, 2017)

En la consola (Shell) de Python se puede observar el valor de la temperatura ambiente al cual está sometido el sensor DS18B20, la imagen 3.17 muestra este resultado.

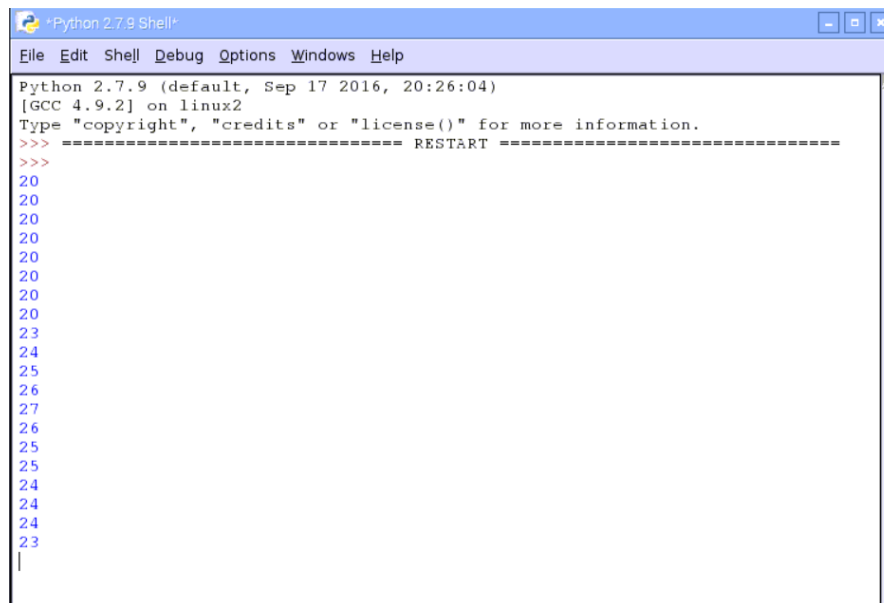


Imagen 3.17 Valores de temperatura en el shell de Python

Fuente: (Python, 2017)

3.1.3 INSTRUCCIONES PARA ENVÍO DE DATOS DE FORMA SERIAL DESDE RASPBERRY PI

Es necesario llamar a la librería necesaria para comunicación serie, para esto escriba en Python lo siguiente:

```
import serial
```

Para otorgar permisos de acceso a la lectura y escritura en la UART y configurar la velocidad de comunicación serie, escriba lo siguiente:

```
os.system('sudo chmod a+rw /dev/ttyAMA0')  
puerto= serial.Serial('/dev/ttyAMA0',baudrate=9600)
```

Estas líneas deben agregarse en el programa anterior, la imagen 3.18 muestra lo descrito.

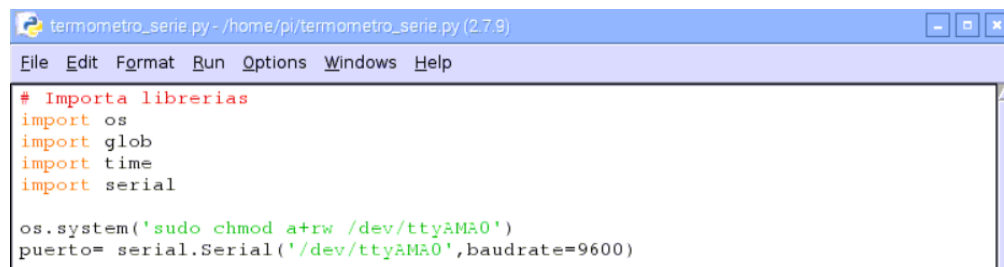


Imagen 3.18 Configuración de la comunicación serie en Python

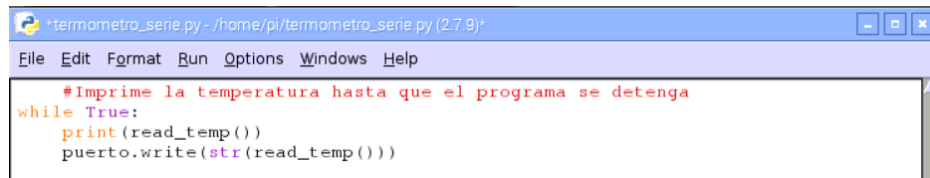
Fuente: (Python, 2017)

Para enviar el valor de temperatura una vez adquirida en forma serie al módulo Bluetooth, en la estructura **while** del programa anterior se agrega la siguiente línea.

```
puerto.write(str(read_temp()))
```

La imagen 3.19, muestra la estructura **while** con las dos líneas que son necesarias para adquirir el valor de temperatura ambiente y enviarla en forma serie hasta el módulo Bluetooth.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



```
*termometro_serie.py - /home/pi/termometro_serie.py (2.7.9)*
File Edit Format Run Options Windows Help
#Imprime la temperatura hasta que el programa se detenga
while True:
    print(read_temp())
    puerto.write(str(read_temp()))
```

Imagen 3.19 Envío de datos de temperatura en forma serial

Fuente: (Python, 2017)

3.2 CREACIÓN DE LA APLICACIÓN MÓVIL EN APP INVENTOR

3.2.1 CREACIÓN DEL DISEÑO DE LA INTERFACE EN APP INVENTOR

Una vez lista la cuenta en MIT App Inventor, se creó una aplicación que permita recibir los datos de temperatura ambiente enviados en forma serial por el microcomputador Raspberry Pi, la aplicación muestra el valor de temperatura, además si la temperatura ambiente está en el rango de 10 a 19 grados centígrados, se mostrara en mensaje “Clima Frío” de color verde, si está de 20 a 24 grados centígrados el mensaje es “Clima Normal” de color azul, cuando está entre 25 a 29 grados centígrados el mensaje es “Clima Caliente” de color rojo, cuando la temperatura ambiente es igual o mayor a 30 grados centígrados, se escucha una alarma en el celular.

Ingresa a MIT App Inventor, cambie al idioma español, una vez en el sitio desde el menú Proyectos, seleccione Comenzar un proyecto nuevo, como se muestra en la imagen 3.20

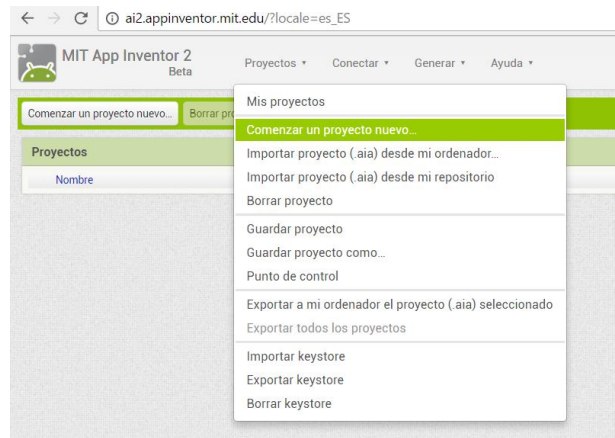


Imagen 3.20 Creación nuevo proyecto en App Inventor

Fuente: (App Inventor, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Aparece una ventana donde pide que escriba un nombre al proyecto, escriba el nombre como el de la imagen 3.21, pulse en el botón Aceptar.

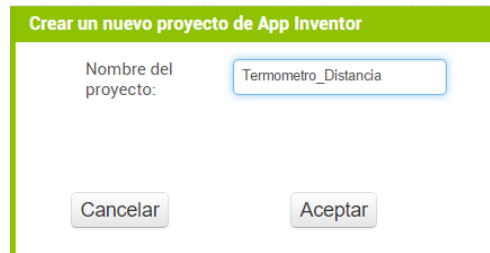


Imagen 3.21 Asignación de nombre al proyecto

Fuente: (App Inventor, 2017)

Cuando pulse el botón Aceptar aparece la interface como muestra la imagen 3.22.

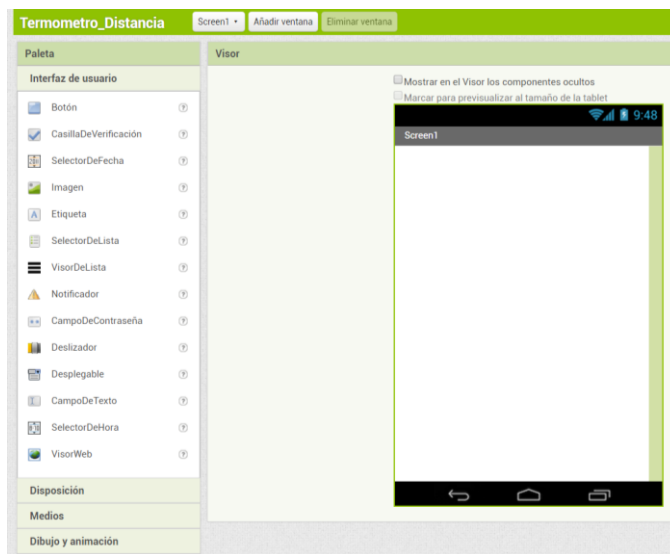
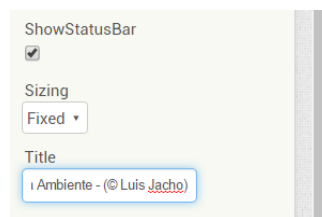


Imagen 3.22 Interface del diseño de App Inventor

Fuente: (App Inventor, 2017)

Seleccione Screen1, en la propiedad **Title** escriba el texto que se visualizará en la barra de título en lugar de Screen1.



“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Imagen 3.23 Asignación de título al proyecto

Fuente: (App Inventor, 2017)

El resultado se visualiza en la imagen 3.23.

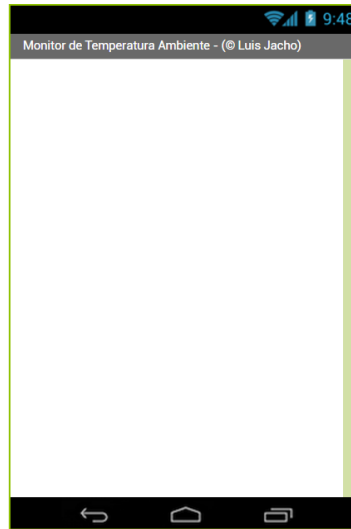


Imagen 3.24 Título de la aplicación

Fuente: (Jacho, 2017)

En la propiedad **DispHorizontal**, seleccione la opción 3, esto centrara todos los componentes que se coloquen en el visor.

En la propiedad **Icono**, suba el icono que desee que aparezca cuando se crea la aplicación y se instale en el dispositivo móvil. La imagen 3.25, muestra todo lo descrito.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

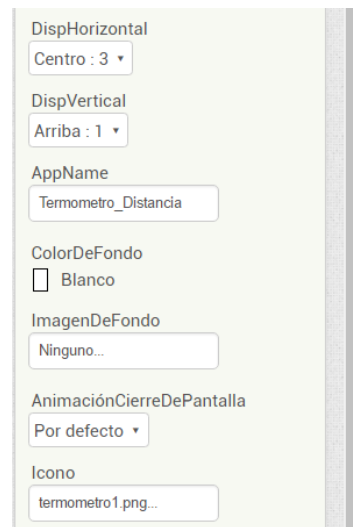


Imagen 3.25 Modificación de la propiedad DispHorizontal e Icono

Fuente: (App Inventor, 2017)

Desde la paleta **Disposición** seleccione **DisposiciónHorizontal** y arrástrelo al visor de la aplicación, dentro de este elemento se colocan los componentes que se mostraran de izquierda a derecha. Seleccione este componente y modifique la propiedad **Ancho** con la opción **Ajustar al contenedor**. La imagen 3.26 muestra el procedimiento.

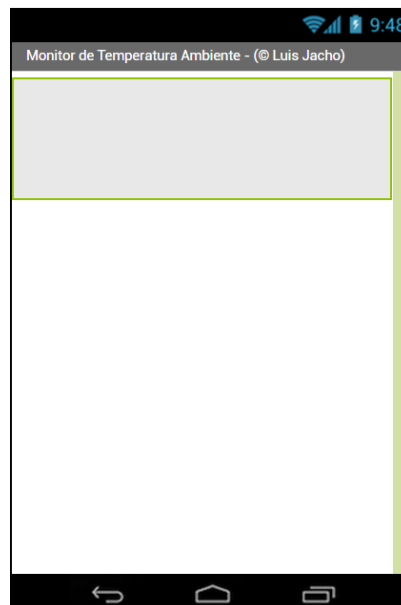


Imagen 3.26 Elemento disposición horizontal

Fuente: (Jacho, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

De la paleta **Interfaz de Usuario**, seleccione **SelectorDeLista**, este componente servirá para que cuando se pulse sobre él, liste los dispositivos Bluetooth que se encuentren emparejados al celular y se conecte a uno de ellos. Arrástrelo dentro del componente disposición horizontal. Escriba en la propiedad **Texto** la palabra Conectar, cambia el nombre del componente a **Conectar**, la imagen 3.27 muestra lo explicado.

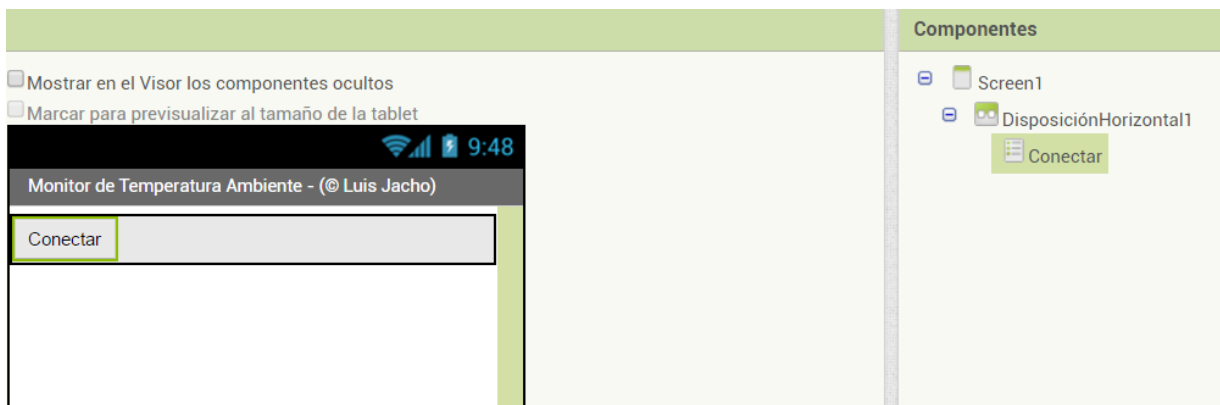


Imagen 3.27 Selección y configuración del componente SelectorDeLista

Fuente: (Jacho, 2017)

De la paleta **Interfaz de Usuario**, seleccione **Botón**, dicho componente servirá para desconectar la conexión Bluetooth, para cambiar los nombres proceda como lo indicado en el componente anterior. La imagen 3.28 muestra el componente disposición horizontal y los dos elementos que lo contiene.

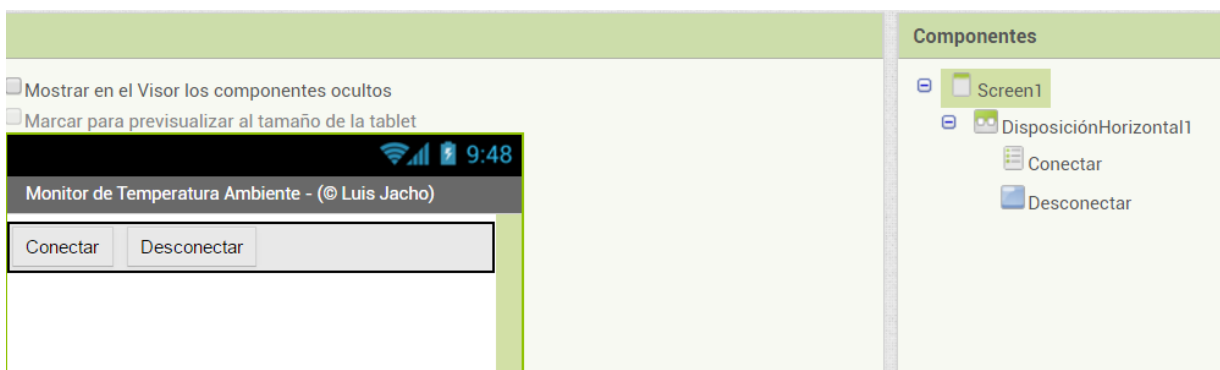


Imagen 3.28 Selección y configuración del componente Botón

Fuente: (Jacho, 2017)

Modifique las siguientes propiedades del componente llamado Conectar:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Propiedad	Opción
Negrilla	Marcado
Tamaño de letra	20.0
Alto	Ajustar al contenedor
Ancho	Ajustar al contenedor
ColorDeTexto	Azul

Cuadro 3.1 Propiedades de diseño del componente Conectar

Las propiedades modificadas del componente Botón se especifican en el cuadro 3.2.

Propiedad	Opción
Habilitado	Desmarcado
Negrilla	Marcado
Tamaño de letra	20.0
Alto	Ajustar al contenedor
Ancho	Ajustar al contenedor
ColorDeTexto	Gris claro

Cuadro 3.2 Propiedades de diseño del componente Botón

Una vez modificadas las propiedades de los componentes, el resultado se observa en la imagen 3.29.

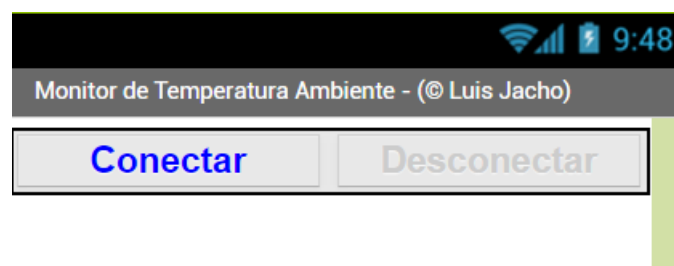


Imagen 3.29 Apariencia del componente Conectar y del Botón

Fuente: (Jacho, 2017)

Seleccione el componente **Imagen** desde la paleta **Interfaz de usuario**, arrastre al visor de la aplicación, desde la propiedad **Foto** cargue la imagen de la universidad previamente descargada, la imagen 3.30 indica lo realizado.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



Imagen 3.30 Imagen agregada al visor de la aplicación

Fuente: (Jacho, 2017)

Coloque en el visor de la aplicación otro componente de Disposición Horizontal, modificando la propiedad **Alto** a 25 pixeles y la propiedad **Ancho** a la opción **Ajustar al contenedor**, en la imagen 3.31 se observa el resultado.

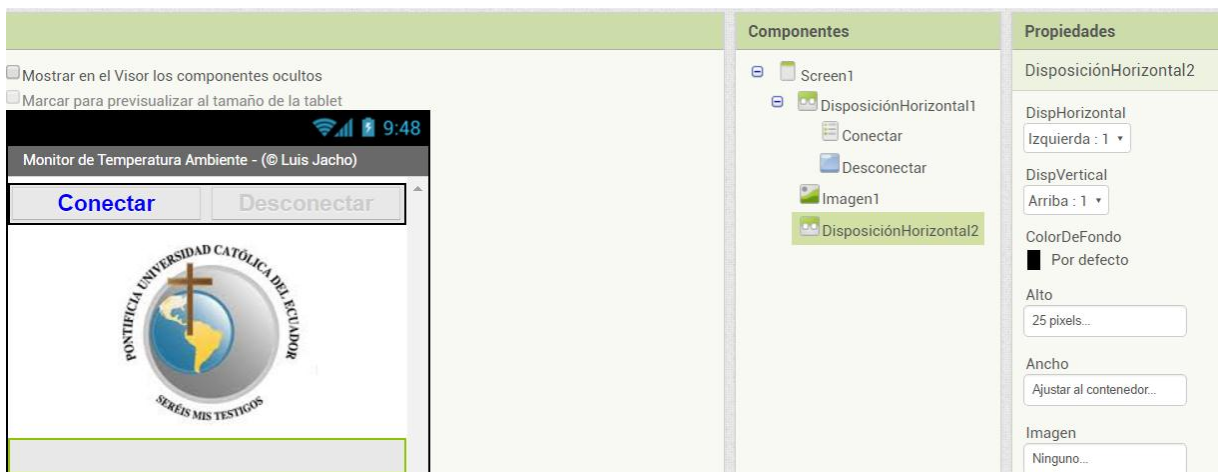


Imagen 3.31 Características del componente DisposiciónHorizontal2

Fuente: (Jacho, 2017)

Coloque tres etiquetas dentro de una disposición horizontal, la primera y la tercera etiqueta no modifique el nombre, la segunda asigne el nombre de **dato**, en esta etiqueta se visualizará el

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

dato de la temperatura ambiente. En la propiedad **Texto** de la primera etiqueta escriba la palabra “**Temperatura=**”, en la segunda borre todo el contenido y en la tercera escriba “**° C**”. El tamaño de letra para las tres etiquetas es 24.0. La imagen 3.32 muestra el resultado de lo explicado en este párrafo.



Imagen 3.32 Etiquetas para visualizar datos de la temperatura

Fuente: (Jacho, 2017)

Ubique otra disposición horizontal, en la propiedad alto escriba 15 pixeles y en la propiedad ancho la opción ajuste al contenedor. El resultado se observa en la imagen 3.33.

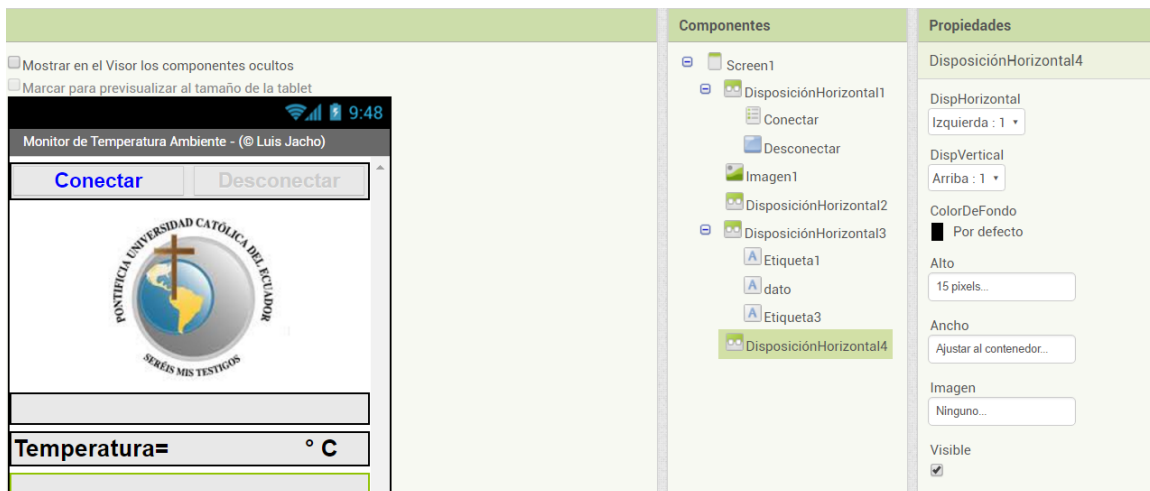


Imagen 3.33 Ubicación de la disposición horizontal 4 en el visor

Fuente: (Jacho, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Dentro de otro contenedor horizontal con un alto de 50 pixeles, coloque otra etiqueta, cámbiele el nombre a **Aviso**, en la propiedad color de fondo, seleccione el color amarillo, el estilo de letra en negrilla y cursiva, el tamaño de letra 28, las propiedades alto y ancho con la opción ajuste al contenedor, la propiedad posición del texto centrado, en la propiedad texto escriba la frase “Aviso del Clima” y en la propiedad color de texto seleccione el verde, en la imagen 3.34 se observan los resultados.

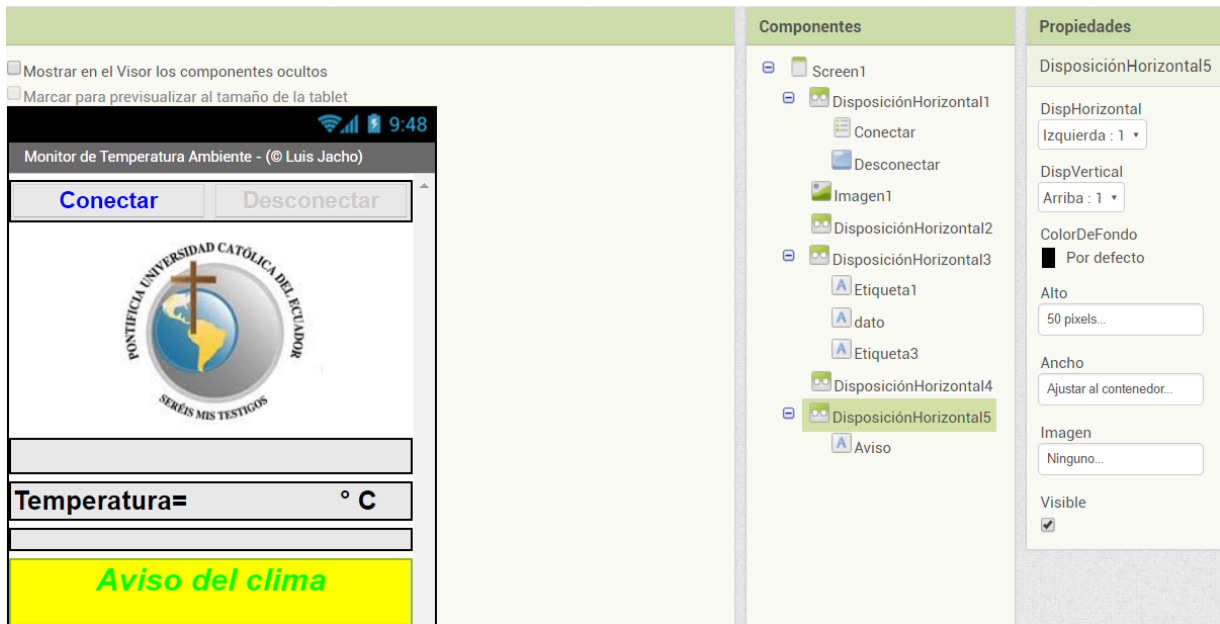


Imagen 3.34 Ubicación de la etiqueta que muestra el tipo de clima

Fuente: (Jacho, 2017)

Para finalizar la colocación de los componentes visibles, ubique otra disposición horizontal con la propiedad alto de 20 pixeles y ancho con la opción ajustar al contenedor. A continuación coloque un botón con un color de fondo naranja, el tamaño de texto de 20, estilo negrilla, con nombre y texto la palabra “Salir”. La imagen 3.35 muestra el visor con todos los componentes visibles de la aplicación.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



Imagen 3.35 Interface de la aplicación para el monitoreo de temperatura

Fuente: (Jacho, 2017)

Los componentes no visibles que se utilizaron en esta aplicación son:

El **CienteBluetooth1** sirve para establecer la comunicación entre el dispositivo móvil y el módulo Bluetooth HC-06.

El componente **Reloj1** sirve para ejecutar el programa cada determinado tiempo, para este proyecto el intervalo de ejecución es de 500 ms, para lo cual se debe colocar en la propiedad IntervaloDelTemporizador un valor de 500.

El componente **Notificador1** sirve para mostrar mensajes en tiempo de ejecución de acuerdo a una condición dada.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

El componente **Sonido1** empleado para reproducir un sonido cuando la temperatura sea igual o superior a 30 grados centígrados.

Estos componentes se encuentran en la parte inferior del visor de la aplicación como se observa en la imagen 3.36.

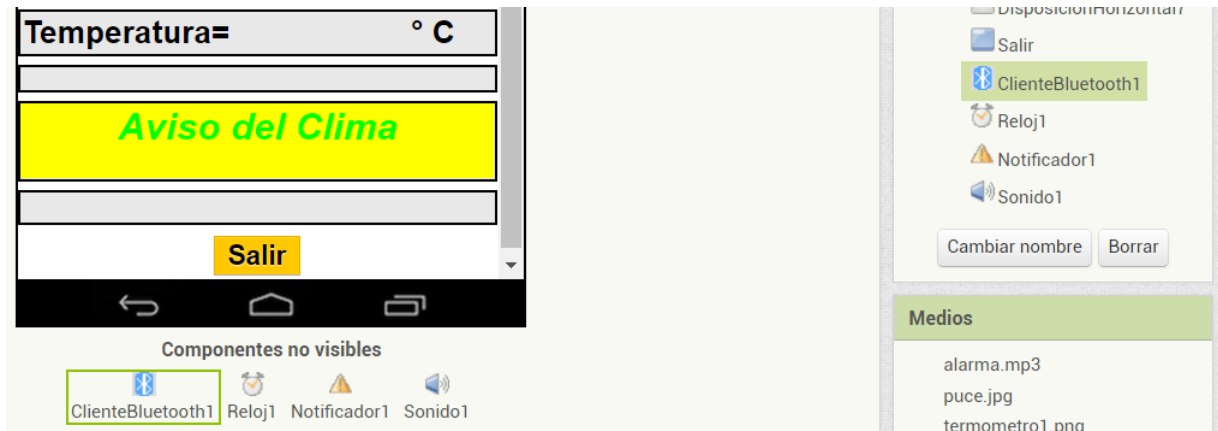


Imagen 3.36 Componentes no visibles de la aplicación

Fuente: (Jacho, 2017)

3.2.2 PROGRAMACIÓN DE LA APLICACIÓN MÓVIL EN APP INVENTOR

Con todos los componentes ubicados en el visor de la aplicación el siguiente paso es programarlos a cada uno para que cumplan la tarea respectiva, la programación en MIT App Inventor es mediante bloques de cada componente y también los que vienen integrados en el desarrollador web. Para acceder a esta interface pulse el botón Bloques que se encuentra en la parte superior derecha de la página web, la imagen 3.37 muestra la pantalla de programación en bloques de MIT App Inventor.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

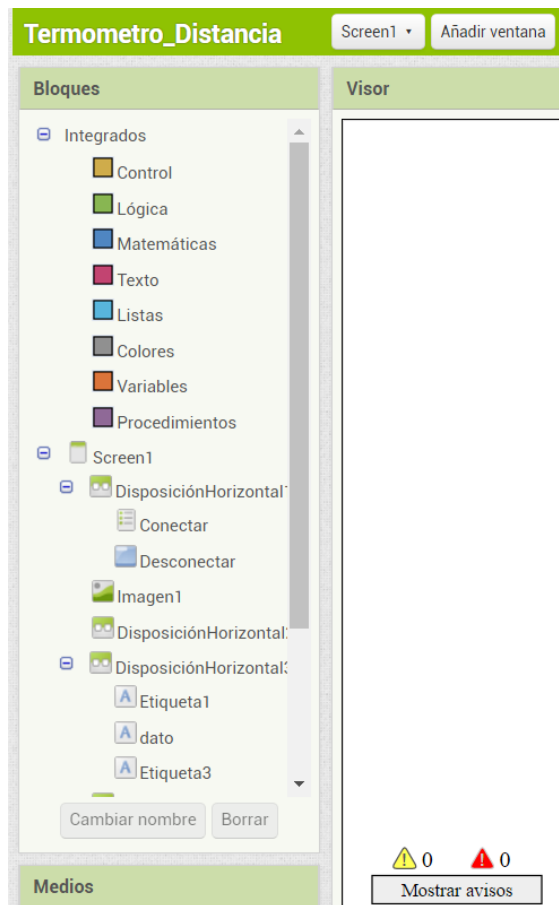


Imagen 3.37 Pantalla para programación en bloques

Fuente: (App Inventor, 2017)

De un clic sobre el bloque Conectar que representa al selector de lista, aparece una ventana que muestra los eventos que tiene cada componente, seleccione el evento que se ejecuta antes de la selección, dentro de éste colocar otros eventos que sirven para listar los clientes Bluetooth que están emparejados con direcciones y nombres para seleccionar el que corresponda, la imagen 3.38 representa lo expuesto.



Imagen 3.38 Evento que lista clientes Bluetooth emparejados

Fuente: (Jacho, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Nuevamente dar un clic sobre el bloque Conectar, seleccione el evento después de la selección, dentro coloque un evento de condición, el mismo que pregunta si se seleccionó algún cliente proceda a conectarlo. Cuando el cliente esté conectado realizar lo siguiente:

- Deshabilita el botón selector de lista llamado Conectar y pone al texto en color gris.
- Habilita el botón llamado Desconectar y pone al texto color rojo.
- Habilita al temporizador para que se ejecute cada 500 ms.
- A través del componente notificador muestra una venta de mensaje cuyo título es Estado de Conexión, el mensaje que presenta es “Dispositivo Conectado” y en la parte inferior de la ventana un botón cuyo texto es Aceptar.

La imagen 3.39 muestra los bloques que realiza lo descrito en el párrafo anterior.

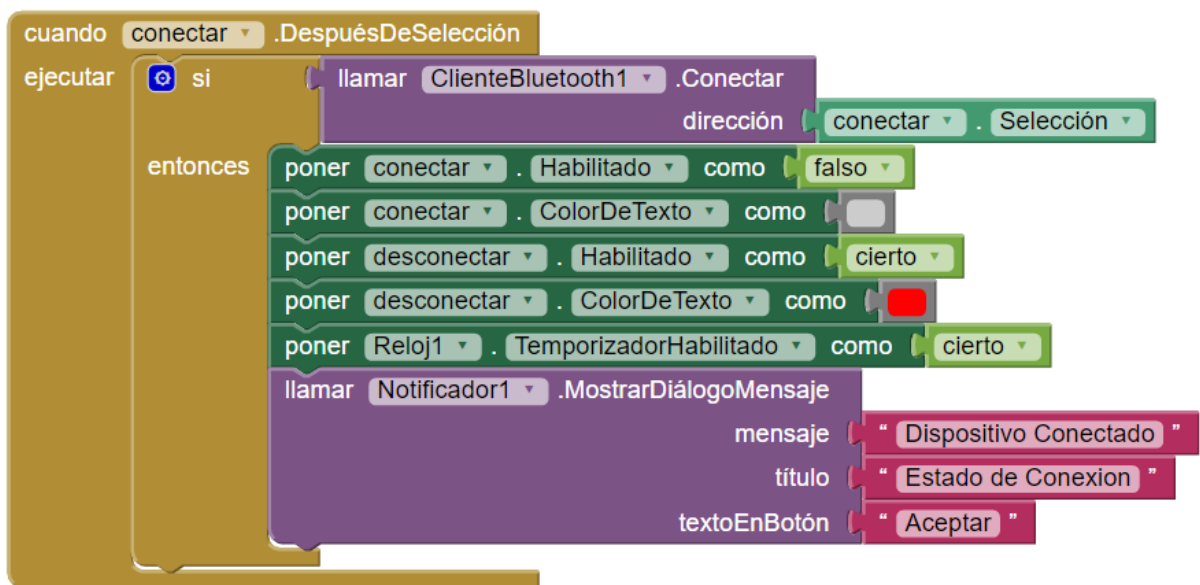


Imagen 3.39 Bloques de programa después de la selección

Fuente: (Jacho, 2017)

El bloque de programación que corresponde al componente Reloj1, contiene primero un evento de comprobación que sirve para verificar si existen bytes para ser recibidos en forma serial; es decir, si el evento **BytesDisponiblesParaRecibir** es mayor que cero. Cuando existen datos se ejecuta un evento que los coloca en el componente llamado **dato**.

También se ejecutan cuatro condiciones:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

- La primera condición pregunta si el dato está entre 10 y 19 grados centígrados, si se cumple coloca el fondo del componente **dato** como verde, escribe en el componente **aviso** la frase Clima Frio con un color verde.
- La segunda condición pregunta si el dato está entre 20 y 24 grados centígrados, si se cumple coloca el fondo del componente **dato** como azul, escribe en el componente **aviso** la frase Clima Normal con un color azul.
- La tercera condición pregunta si el dato está entre 25 y 29 grados centígrados, si se cumple coloca el fondo del componente **dato** como rojo, escribe en el componente **aviso** la frase Clima Caliente con un color rojo.
- La cuarta condición pregunta si el dato es igual o mayor que 30 grados centígrados, si se cumple reproduce un sonido.

La imagen 3.40 se visualizan los bloques de programa que están dentro del bloque Reloj1.

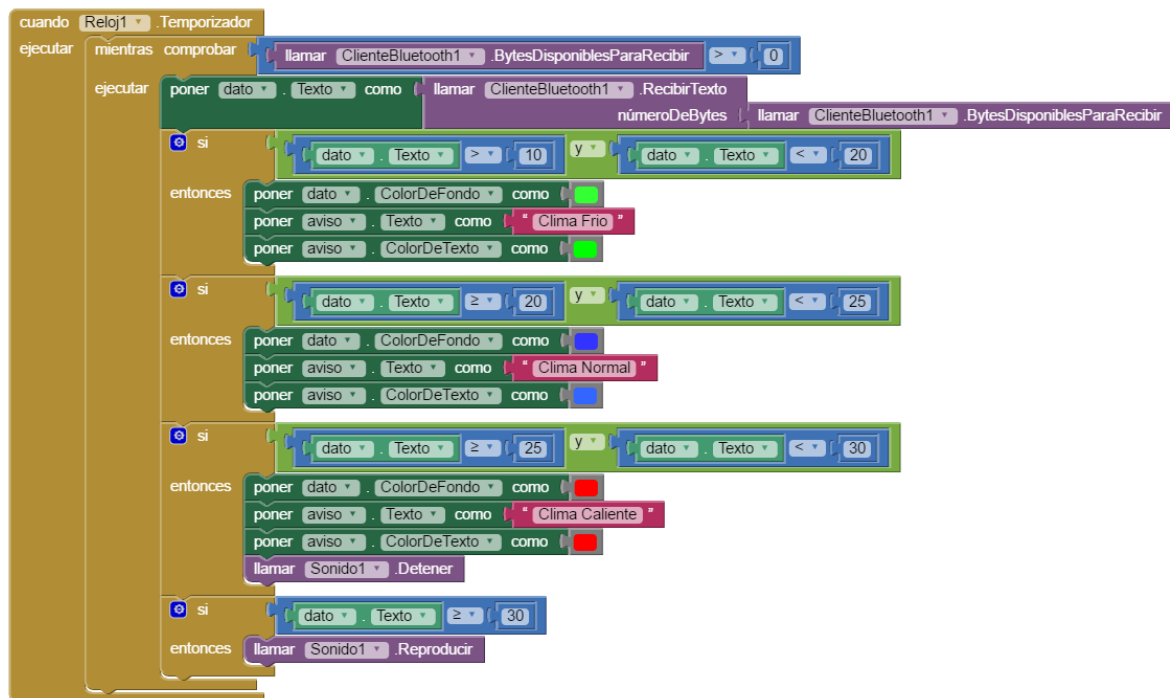


Imagen 3.40 Bloque que se ejecutan cada 500 ms

Fuente: (Jacho, 2017)

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Cuando se pulsa el botón Desconectar, se ejecuta lo siguiente:

- Se habilita el selector de lista y se coloca el texto de color azul.
- Se deshabilita el botón desconectar y se coloca el texto de color gris.
- Se deshabilita el temporizador.
- En el componente dato se borra el texto y el fondo cambia a blanco.
- En el componente aviso escribe el texto “Aviso del Clima”.
- El cliente se desconecta.
- Aparece la venta de dialogo de mensaje, el título de esta es Estado de Conexión, el mensaje Dispositivo Desconecto y un botón Aceptar.

La imagen 3.41 muestra la programación en bloques el botón Desconectar.



Imagen 3.41 Bloques de programa botón Desconectar
Fuente: (Jacho, 2017)

Cuando se pulsa el botón Salir se ejecuta un evento que cierra la aplicación, este indica la imagen 3.42.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

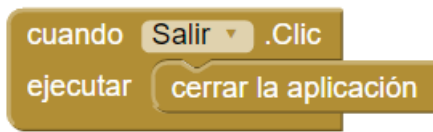


Imagen 3.42 Bloque de programa que finaliza la aplicación

Fuente: (Jacho, 2017)

Para crear la aplicación y descárgala en el computador para luego transferir al dispositivo móvil, se da un clic en el menú Generar y seleccione **App (guardar archivo .apk en mi ordenador)**. La imagen 3.43 visualiza el procedimiento descrito.

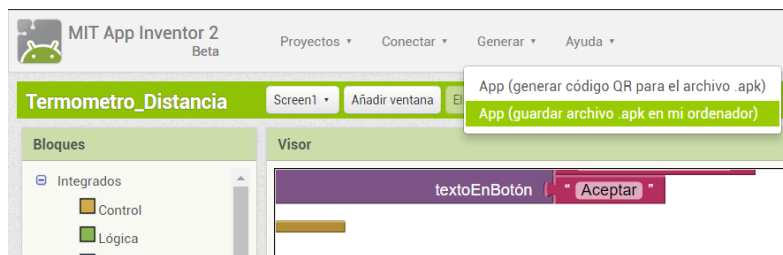


Imagen 3.43 Generación aplicación

Fuente: (Jacho, 2017)

3.3 PRUEBAS DE FUNCIONAMIENTO DE LA APLICACIÓN MÓVIL PARA EL MONITOREO DE TEMPERATURA AMBIENTE

Para comprobar el funcionamiento de la aplicación es necesario realizar el siguiente procedimiento:

- Transfiera la aplicación desde el computador al dispositivo móvil.
- Encienda el microcomputador Raspberry Pi.
- Realice el emparejamiento entre el dispositivo móvil y el modulo Bluetooth HC-06.
- Instale la aplicación en el dispositivo móvil.
- Acceda en forma remota al escrito del Raspbian desde Windows.
- Abra la aplicación instalada en el dispositivo móvil (ver imagen 3.44).
- De clic en el botón conectar, aparece otra ventana donde lista los dispositivos Bluetooths emparejados, selección el que corresponde al módulo HC-06. Si los dispositivos se conectaron el led que estaba parpadeando en el módulo HC-06 se encenderá de manera fija.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

- Desde Raspbian abra el programa realizado para adquirir el valor de la temperatura ambiente y enviarlo mediante Bluetooth al dispositivo móvil, ejecute el mismo, esto se puede verificar si en la consola del Python se muestran los valores de temperatura.



Imagen 3.44 Aplicación abierta sin conexión

Fuente: (Jacho, 2017)

Una vez conectados los dispositivos y ejecutado el programa en el Raspberry en la imagen 3.45 se presenta una adquisición de 24 grados centígrados, en este valor de acuerdo al programa se debe visualizar en el componente **dato**, el texto de fondo azul y en el componente **aviso** el mensaje Clima Normal

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



Imagen 3.45 Aplicación mostrando valor de temperatura

Fuente: (Jacho, 2017)

CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- Para el funcionamiento del microcomputador Raspberry Pi es necesario instalar un software como el Raspbian para que pueda ejecutar tareas específicas, las mismas que son programadas en Python.
- Se asignó una IP estática al Raspberry Pi con el propósito de tener acceso remoto al escritorio de Raspbian desde Windows mediante el software TightVNC Viewer y de esta manera evitar el uso de dispositivos adicionales como teclado, mouse y TV.
- Raspberry Pi no puede adquirir señales analógicas, es por esta razón que se empleó el sensor de temperatura digital DS18B20 que tiene un protocolo de comunicación One Wire.
- Para no tener problemas en la comunicación serie del Raspberry Pi, es necesario habilitar mediante comando de sistema los permisos respectivos para que se pueda realizar la lectura y escritura en el puerto.
- Se debe agregar una línea de comando al final del archivo de configuración del Raspberry Pi para habilitar la comunicación con el DS18B20 mediante el protocolo One Wire.
- El Raspberry Pi envía los datos de la temperatura ambiente al dispositivo móvil a través del módulo Bluetooth HC-06, el mismo que debe emparejarse para que exista la comunicación con el dispositivo móvil.
- La aplicación para el dispositivo móvil se realizó en MIT App Inventor, esta visualiza la temperatura ambiente enviada por el Raspberry variando los colores de fondo del componente que muestra el valor de temperatura en una rango de 0 a 29 grados centígrados y cuando supera el valor máximo del rango se escucha un sonido que representa peligro.
- App Inventor es una herramienta web de desarrollo para aplicaciones que se ejecutarán en los dispositivos móviles con sistema operativo Android.

4.2 RECOMENDACIONES

- Alimentar el microcomputador Raspberry Pi con una fuente de voltaje de 5 voltios de corriente continua mínimo 2 amperios.
- Realizar el emparejamiento entre en dispositivo móvil y el módulo Bluetooth HC-06, caso contrario no se listará al momento de pulsar el botón conectar de la aplicación.
- No exceder la distancia máxima de 10 metros que recomienda el fabricante del módulo HC-06, con esto evitará que se pierda la comunicación.
- Descargar la aplicación móvil mediante lectura de código QR que provee App inventor, para una sencilla instalación.
- Descargar NOOBS para tener opciones al momento de elegir el sistema operativo que se va a instalar.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

REFERENCIAS

- Amaya Balaguera, Y. D. (14 de Noviembre de 2013). *Metodologías ágiles en el desarrollo de aplicaciones*. Colombia.
- Borrego, O. (16 de 10 de 2015). *¿Que es Kodi? No preguntes, solo instalalo!* Recuperado el 11 de 08 de 2016, de MUNDO KODI: <http://mundokodi.com/que-es-kodi/#gs.AGt=PvA>
- Castellano, A. R. (2012). *Bluetooth introducción a su funcionamiento*. Madrid, España.
- Enterprise, R. H. (2009). *Protocolo SSH*. Obtenido de <http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ssh.html>
- Fabrica Digital. (3 de enero de 2016). *Empezando a programar en Raspberry Pi con Python*. Obtenido de <https://fabricadigital.org/2016/01/hola-pi-empezando-a-programar-en-raspberry-pi-con-python/>
- Fundación Raspberry. (s.f.). Recuperado el 2 de Octubre de 2016, de Raspberry Pi 1 Modelo A: <https://www.raspberrypi.org/products/model-a/>
- Fundación Raspberry . (s.f.). *Raspberry Pi 1 Modelo A+*. Recuperado el 2 de Octubre de 2016, de <https://www.raspberrypi.org/products/model-a-plus/>
- Fundacion Raspberry Pi. (s.f.). *Preguntas frecuentes*. Recuperado el 2 de Octubre de 2016, de <https://www.raspberrypi.org/help/faqs/#introWhatIs>
- Fundacion Raspberry. (s.f.). *Python*. Recuperado el 15 de Noviembre de 2016, de <https://www.raspberrypi.org/documentation/usage/python/>
- Fundacion Raspberry. (s.f.). *Raspberry Pi 1 Modelo B*. Recuperado el 2 de Octubre de 2016, de <https://www.raspberrypi.org/products/model-b/>
- Fundación Raspberry. (s.f.). *Raspberry Pi 1 Modelo B+*. Recuperado el 2 de Octubre de 2016, de <https://www.raspberrypi.org/products/model-b-plus/>
- Fundación Raspberry. (s.f.). *Raspberry Pi 2 Modelo B*. Recuperado el 2 de Octubre de 2016, de <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
- Fundación Raspberry. (s.f.). *Raspberry Pi 3 Modelo B*. Recuperado el 2 de Octubre de 2016, de <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- naylampmechatronics. (s.f.). *Tutorial sensor digital de temperatura DS18B20*. Obtenido de http://www.naylampmechatronics.com/blog/46_Tutorial-sensor-de-temperatura-DS18B20.html
- Nergiza. (s.f.). *Raspberry Pi 2 B*. Recuperado el 19 de Octubre de 2016, de <http://nergiza.com/raspberry-pi-2-b-que-es-y-para-que-nos-puede-servir/>
- Pastor, J. (12 de 08 de 2015). *Qué puedes y qué no puedes hacer con Windows 10 IoT Core en las Raspberry Pi 2*. Recuperado el 11 de 08 de 2016, de Xataka: <http://www.xataka.com/makers/que-puedes-y-que-no-puedes-hacer-con-windows-10-iot-core-en-las-raspberry-pi-2>
- Raul C. (2015). *Guía de iniciación a APP INVENTOR*. Madrid, España.
- Rederjo, J. L. (20 de Febrero de 2013). *Uso del AppInventor*. Obtenido de <http://recursostic.educacion.es/observatorio/web/en/software/programacion/1090-uso-de-appinventor-en-la-asignatura-de-tecnologias-de-la-comunicacion-y-la-informacion>
- Redes Zone. (29 de Mayo de 2014). *Raspberry Pi*. Obtenido de <http://www.redeszone.net/raspberry-pi/>

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Sevilla Martín, J. P., & García Sánchez, P. (2006). Comunicación entre dispositivos Bluetooth. España.

Tatham, S. (2011). *PuTTY*. Obtenido de <http://es.ccm.net/download/descargar-395-putty>

Vega, J. I. (Octubre de 2014). *Monitoreo de temperatura y humedad en un campo de cultivo utilizando el*. Jalisco, Mexico.

ANEXOS

Anexo 1: Programa en Python

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

```
# Importa librerias
import os
import glob
import time
import serial

os.system('sudo chmod a+rw /dev/ttyAMA0')
puerto= serial.Serial('/dev/ttyAMA0',baudrate=9600)

# Inicializa los pines GPIO
os.system('modprobe w1-gpio') #Activa el modulo GPIO
os.system('modprobe w1-therm') #Activa el modulo de temperatura

#Encuentra el archivo de dispositivo correcto que contiene datos de temperatura
base_dir= '/sys/bus/w1/devices/'
device_folder= glob.glob(base_dir + '28*')[0]
device_file= device_folder + '/w1_slave'

#Una funcion que lee datos del sensor
def read_temp_raw():

    f= open(device_file,'r') #Abre el archivo del dispositivo de temperatura
    lines= f.readlines() #Devuelve el texto
    f.close()

    return lines

#Convierte el valor del sensor en temperatura
def read_temp():
    lines= read_temp_raw() #Lee la temperatura del archivo de dispositivo
```

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA
TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA
RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

```
#Mientras que la primera linea no contenga 'YES', espera 0,2 s
#y luego lee el archivo de dispositivo de nuevo
while lines[0].strip()[-3:] != 'YES':
    time.sleep(0.2)
    lines= read_temp_raw()

#Busca la posicion del '=' en la segunda linea del archivo del dispositivo
equals_pos= lines[1].find('t=')

#Si se encuentra '=' convierte el resto de la linea despues de '='
#en grados centigrados

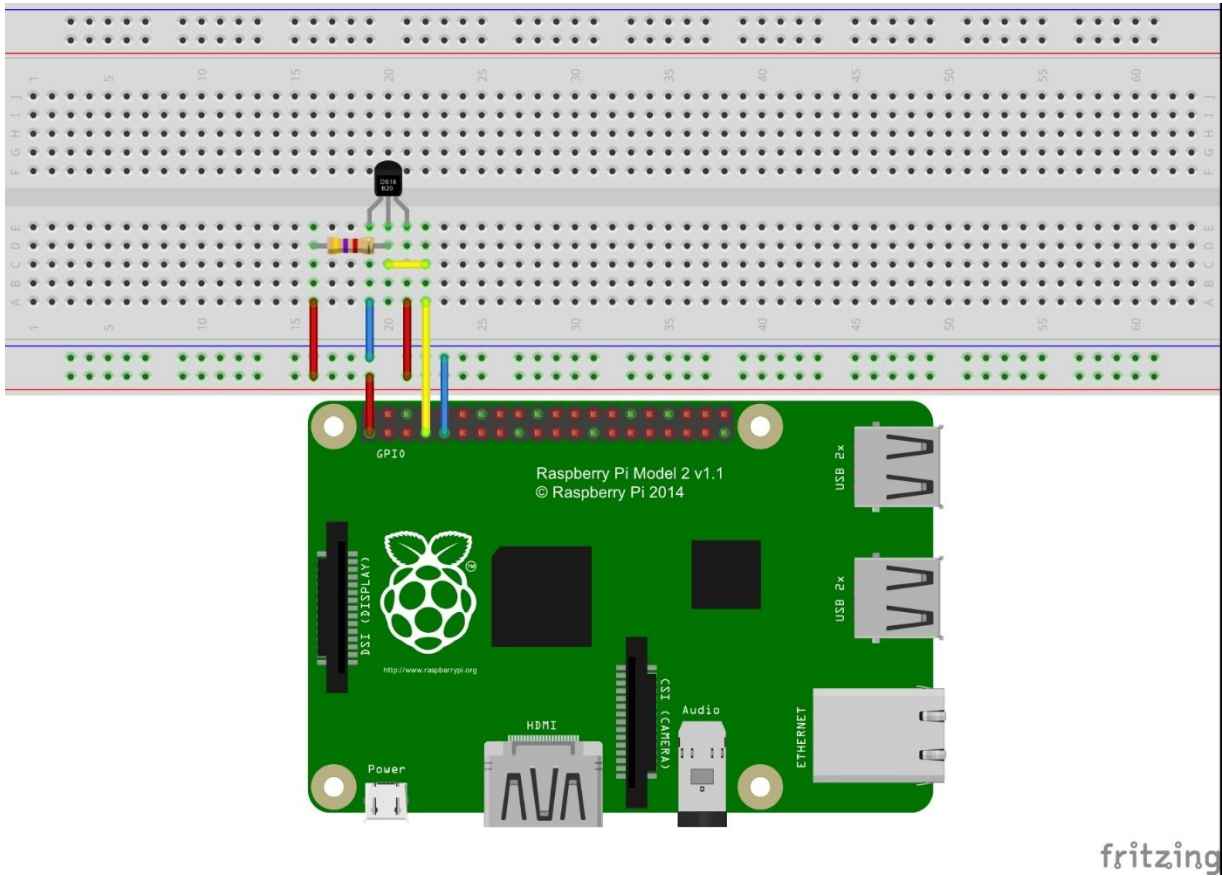
if equals_pos != -1:
    temp_string= lines[1][equals_pos+2:]
    temp_c= float(temp_string)/1000.0
    temperatura=int(temp_c)

return temperatura

#Imprime la temperatura hasta que el programa se detenga
while True:
    print(read_temp())
    puerto.write(str(read_temp()))
```

Anexo 2: Conexión del sensor DS18B20 al Raspberry Pi

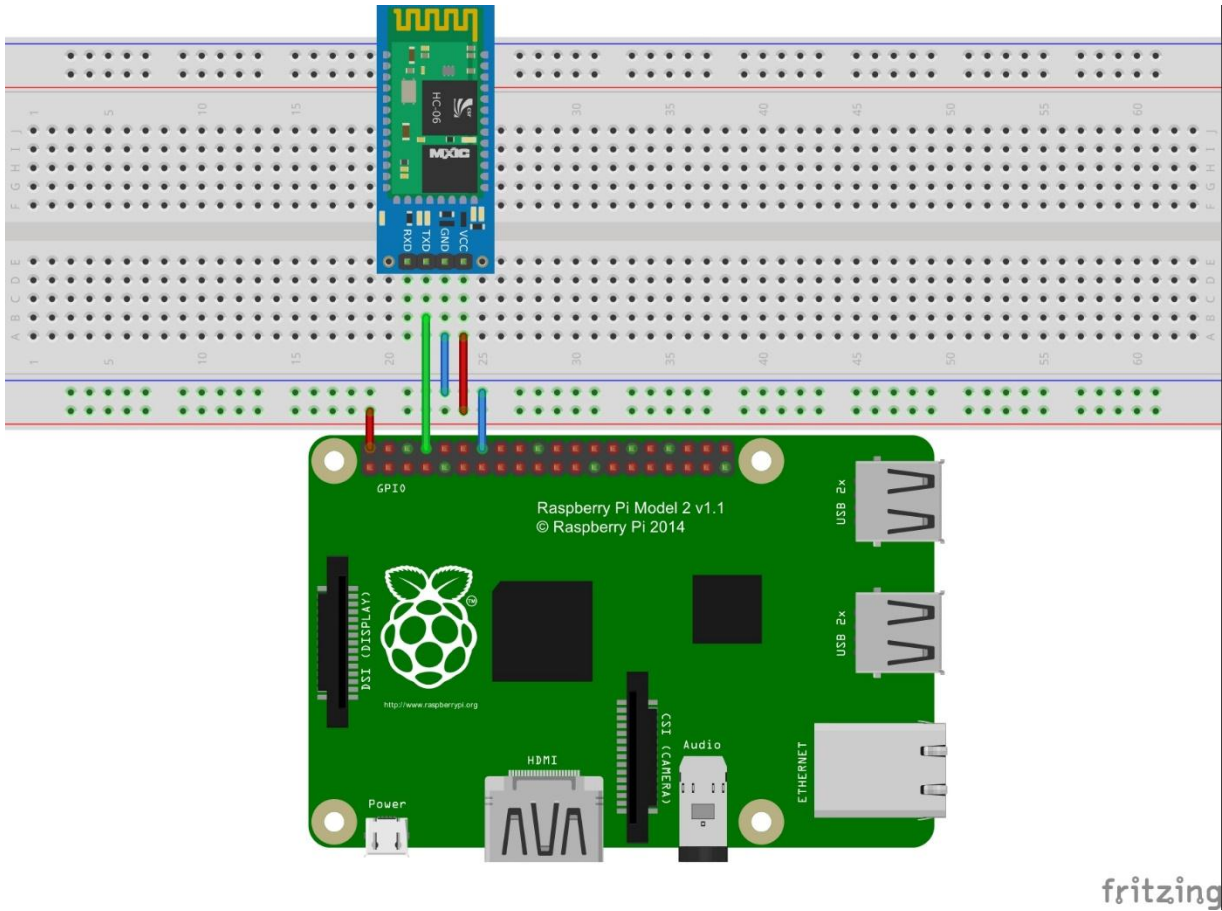
“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



fritzing

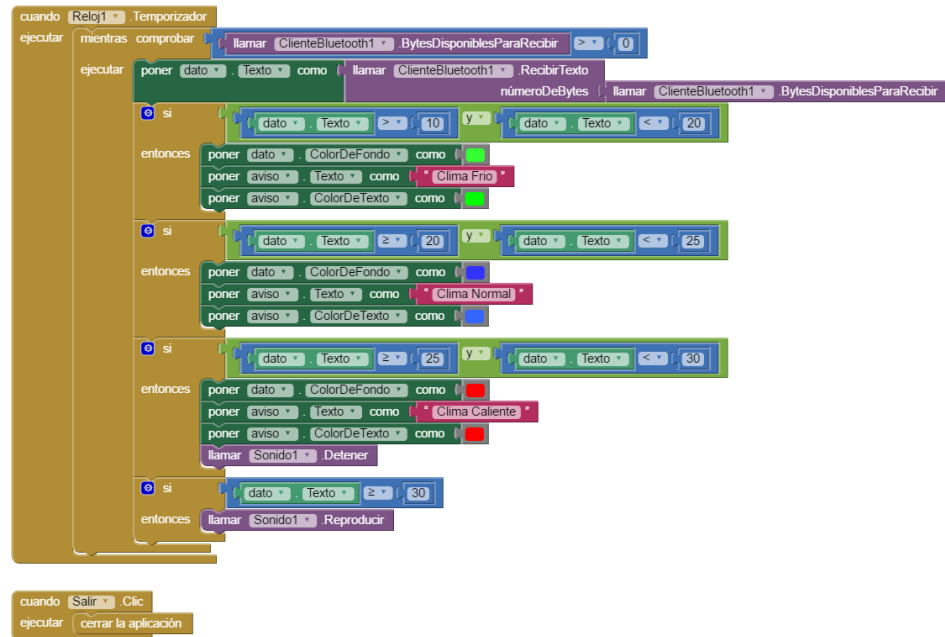
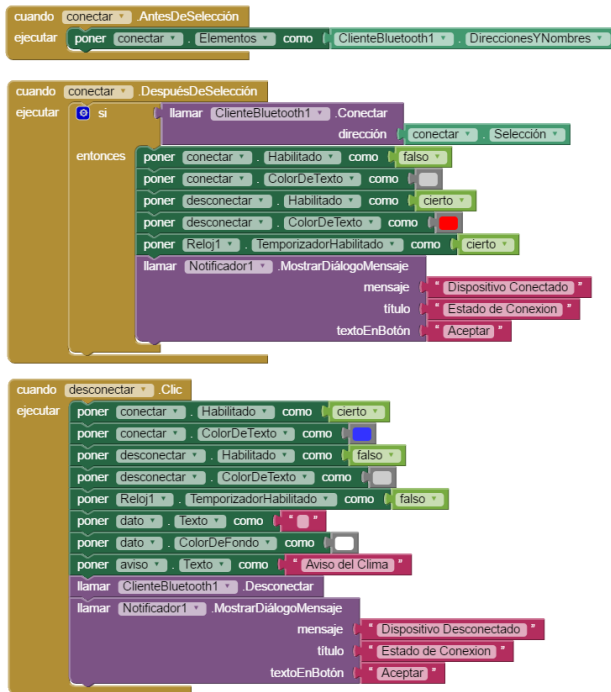
Anexo 3: Conexión del módulo HC-06 al Raspberry Pi

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Anexo 4: Programación en MIT App Inventor



“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

Anexo 4: Manual de usuario

Para el uso correcto de la aplicación móvil del monitoreo de temperatura ambiente siga los siguientes pasos:

1. Copie el archivo .apk a la tarjeta de memoria del dispositivo móvil.
2. Instale la aplicación en el dispositivo móvil
3. Instale el software TightVNC Viewer en un computador con sistema operativo Windows.
4. Mediante el centro de redes y recursos compartidos de Windows asigne la dirección la dirección IP 192.168.0.8 al computador, figura 1.

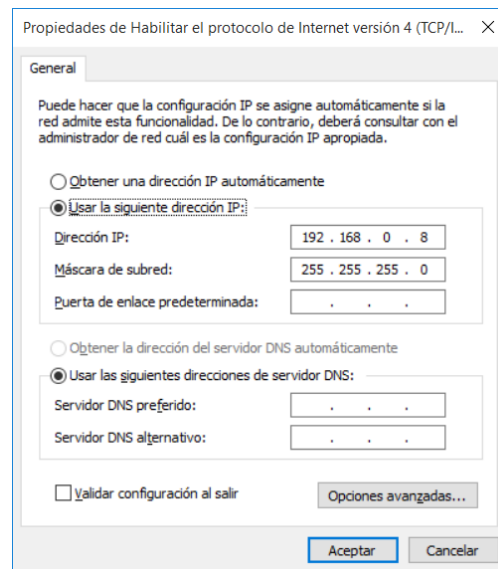


Figura 1

5. Con un cable de red conecta la tarjeta Raspberry Pi con el puerto Ethernet del computador
6. Energice la tarjeta Raspberry con la fuente de 5 V que viene incluida, espere unos segundos hasta que inicie el microcomputador.
7. Abra TightVNC Viewer y escriba la dirección IP estática asignada al Raspberry, seguido de los dos puntos y el número, esto especifica que es el dispositivo número uno, 192.168.0.25:1, figura 2.

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”

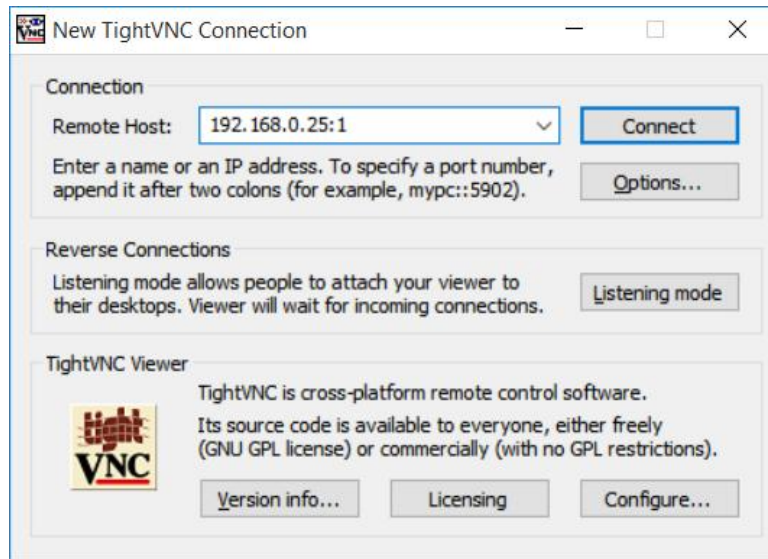


Figura 2

8. Una vez escrita la dirección IP del equipo remoto, pulse en el botón Connect, aparece una ventana autenticación donde pide la contraseña de acceso al dispositivo, escriba *raspberry* en la casilla Password y pulse el botón OK.

9. Desde el menú de Raspbian, en la opción programación abra Python 2, figura 3

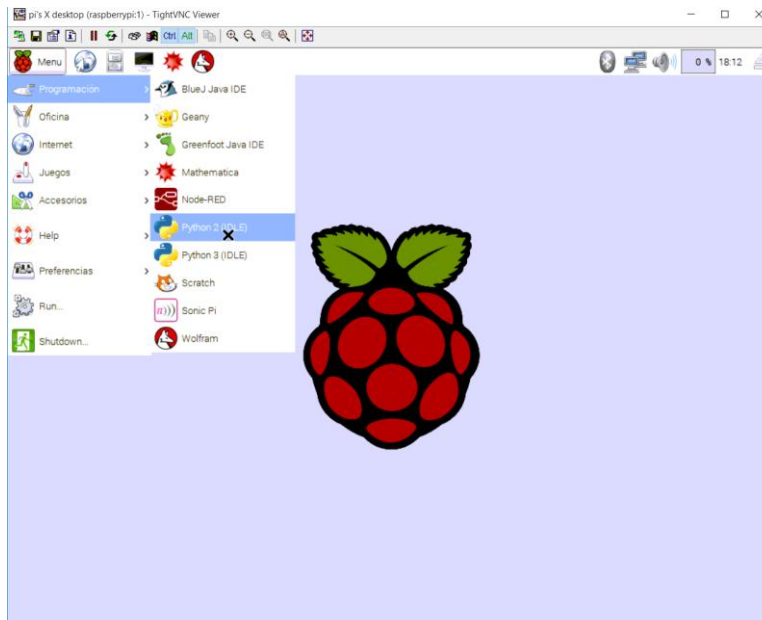
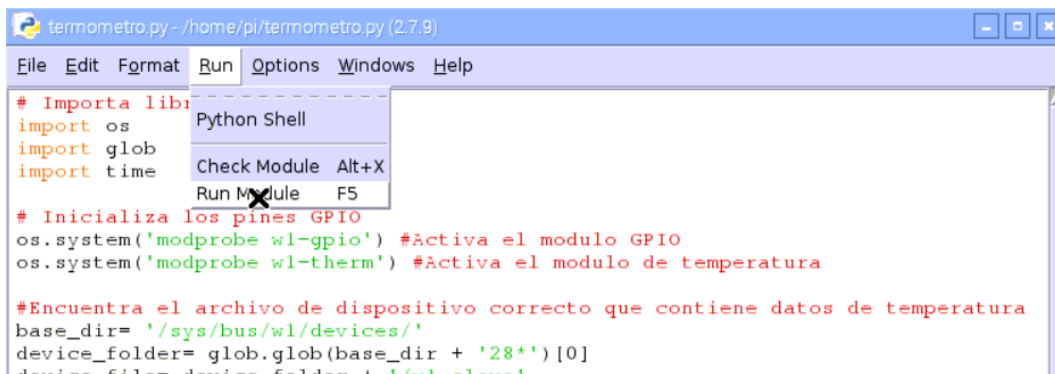


Figura 3

10. Abra el archivo llamado termómetro.py, ejecute el archivo desde el menú Run, ver figura 4

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DESTINADO A MONITOREAR LA TEMPERATURA AMBIENTE, INTEGRANDO COMPUTADOR DE PLACA ÚNICA RASPBERRY PI, CON TECNOLOGÍA MÓVIL.”



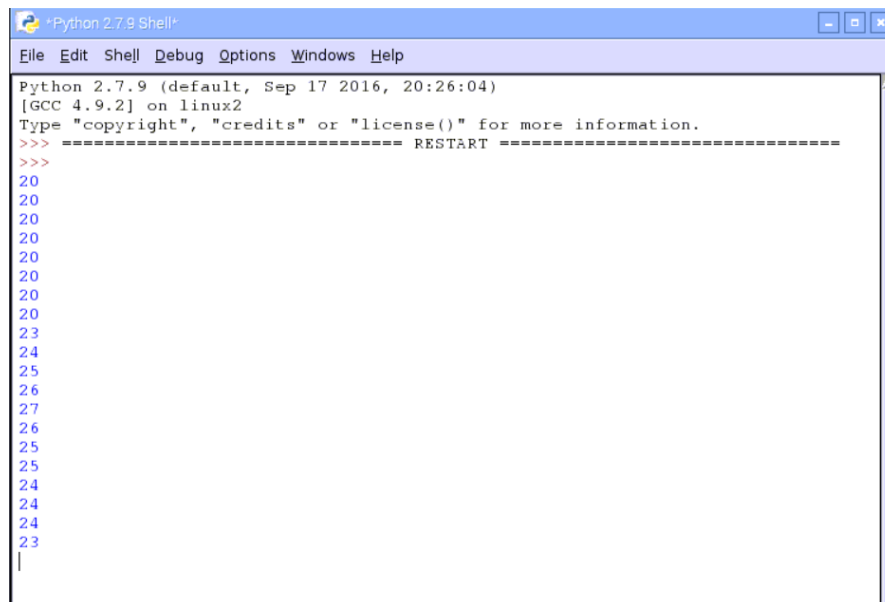
```
# Importa librerías
import os
import glob
import time

# Inicializa los pines GPIO
os.system('modprobe wl-gpio') #Activa el modulo GPIO
os.system('modprobe wl-therm') #Activa el modulo de temperatura

#Encuentra el archivo de dispositivo correcto que contiene datos de temperatura
base_dir= '/sys/bus/wl/devices/'
device_folder= glob.glob(base_dir + '28*')[0]
device_file= device_folder + '/w1_slave'
```

Figura 4

11. Observe que los datos de temperatura aparecen en la consola de python, ver figura 5.



```
Python 2.7.9 (default, Sep 17 2016, 20:26:04)
[GCC 4.9.2] on linux2
Type "copyright", "credits" or "license()" for more information.
>>> ----- RESTART -----
>>>
20
20
20
20
20
20
20
20
23
24
25
26
27
26
25
25
24
24
24
23
```

Figura 5

12. Active el bluetooth del dispositivo móvil y emparejelo con el dispositivo HC-06 conectado Raspberry PI.
13. Ejecute la aplicación instalada en el dispositivo móvil.
14. Pulse el Conectar, aparece un ventana donde lista los dispositivos emparejados, seleccione el HC-06.
15. Observe los datos de temperatura ambiente en la pantalla del dispositivo móvil.
16. Para salir de la aplicación pulse el botón Desconectar y luego Salir.