

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA
CARRERA DE ESPECIALIZACIÓN EN SISTEMAS
EMBEBIDOS



MEMORIA DEL TRABAJO FINAL

Control de acceso biométrico

Autor:
Ing. Christian Yáñez

Director:
Esp.Ing. Ernesto Gigliotti

Jurados:
Esp. Ing. Diego Fernández(FIUBA)
Esp. Ing. Danilo Zecchin (FIUBA)
Esp. Ing. Carlos Mancón(FIUBA)

Este trabajo fue realizado en las Ciudad Autónoma de Buenos Aires, entre mayo de 2018 y diciembre de 2018.

Resumen

En el siguiente documento se describe el desarrollo de un sistema de control de acceso biométrico implementado mediante la utilización de Linux sobre una plataforma de desarrollo embebida.

Dicho sistema cuenta con el diseño de una interfaz gráfica sobre una pantalla con tecnología touch.

El sistema permite a los usuarios tener permisos de acceso en caso de un reconocimiento efectivo de la huella digital. El permiso de acceso se implementa mediante un mensaje sobre la interfaz gráfica y la activación temporal de una salida digital.

Por otro lado, un único usuario reconocido como privilegiado tiene la posibilidad de realizar configuraciones en el sistema de usuarios y acceder de forma remota a un historial de accesos previos.

El proyecto nace como un propósito personal del autor mediante el cual se aplican varias de las herramientas aprendidas durante la carrera de especialización entre las cuales se destacan: sistemas operativos de propósito general (kernel de linux, procesos y threads, gestión de procesos, comunicación entre procesos, Sockets), protocolos de comunicación (periférico UART), programación de micro controladores (Programación orientada a eventos, software modular, diseño de máquinas de estado) entre las mas importantes.

Agradecimientos

A mi madre Martha, por su valioso ejemplo y apoyo incondicional.

A mi director, por su guía, consejo y ayuda.

A mis profesores y compañeros, por su colaboración apoyo y paciencia.

Índice general

Resumen	III
1. Introducción General	1
1.1. Sistemas de control de acceso biométrico	1
1.2. Motivación	2
1.3. Objetivos y alcance	2
2. Introducción Específica	5
2.1. Planteamiento	5
2.1.1. Descripción funcional	5
2.2. Requerimientos	6
2.3. Planificación	7
3. Diseño e Implementación	9
3.1. Plataforma de desarrollo	9
3.1.1. Linux embebido	10
3.1.2. Pantalla táctil element 14	10
3.2. Desarrollo de interfaz gráfica	12
3.2.1. GTK+	12
3.2.2. Glade	12
3.3. Módulo sensor para adquisición de huellas	13
3.3.1. Lectores ópticos reflexivos	13
3.3.2. Módulo lector de huella Adafruit ZFM-20	13
3.4. Sistema General	15
3.5. Subsistema GUI (logic Service)	15
3.6. Subsistema para control del sensor de huella	15
3.7. Subsistema data-web	15
4. Ensayos y Resultados	17
4.1. Pruebas funcionales del hardware	17
5. Conclusiones	19
5.1. Conclusiones generales	19
5.2. Próximos pasos	19

Índice de figuras

1.1.	Aplicaciones con sistemas biométricos	1
1.2.	Disposición de hardware del sistema.	2
1.3.	Módulos funcionales del sistema.	3
2.1.	Secuencia funcional del sistema.	6
2.2.	Diagrama Activity on Node.	8
3.1.	Raspberry Pi 3 B+ módulos constitutivos.	10
3.2.	Interfaz Raspbian Lite.	11
3.3.	Pantalla touch 7 pulgadas element 14 y accesorios para conexión. .	11
3.4.	Entorno de desarrollo Glade.	12
3.5.	Procesamiento de huella dactilar.	14
3.6.	Formato para envío/recepción de paquetes de datos.	16

Índice de Tablas

3.1. caption corto	15
3.2. caption corto	15

Dedicado a Martha Flores

Capítulo 1

Introducción General

1.1. Sistemas de control de acceso biométrico

Partiendo de la definición de biometría como la toma de medidas estandarizadas para el reconocimiento de personas basado en una o mas características físicas o conductuales, se puede definir entonces a los sistemas de control basados en biometría como sistemas que utilizan dichas características como fuente de información que posteriormente a ser analizadas servirán para determinar el comportamiento del mencionado sistema, en el caso particular de este proyecto, determinarán permisos de acceso para el usuario. Existen dos tipos básicos de biometría; la de comportamiento, dentro de la cual se analizan características como la forma de caminar u ondas cerebrales y por otro lado la biometría fisiológica dentro de la cual se cubren los análisis de huella digital, rasgos faciales, forma de manos, retina, firma, vena, voz y análisis genético. De los análisis mencionados, el más popular por su gran nivel precisión en cuanto a reconocimiento es el análisis de huella digital. El uso de los rasgos físicos intrínsecos de las personas como herramienta para su reconocimiento en sistemas de control es en la actualidad uno de los métodos más seguros y su implementación va en aumento. Los sistemas biométricos evitan el tráfico de passwords o tarjetas de identificación que son métodos tradicionales y con gran deficiencia en cuanto a seguridad. La utilización de la tecnología biométrica brinda no solo la posibilidad de implementar controles de acceso, son también utilizados para el reconocimiento en sistemas gubernamentales, transacciones bancarias, servicios de salud entre los más importantes. La figura 1.1” ilustra las potencialidades de los sistemas de acceso biométricos.



FIGURA 1.1: Aplicaciones con sistemas biométricos

1.2. Motivación

Aplicar los conocimientos adquiridos durante la carrera para desarrollar un producto tecnológico enfocado en cubrir necesidades puntuales.

Finalizar exitosamente la carrera mencionada e involucrarse con los procesos de diseño y construcción de sistemas Embebidos para una futuro emprendimiento personal.

Desarrollar un prototipo base de identificación biométrica que a futuro sea implementado en sistemas de vigilancia escolares o domiciliarios.

Implementar un sistema de alta tecnología que represente una opción frente a los módulos comerciales.

Si bien la principal motivación del presente trabajo se enfoca en el orden de lo académico, se pretende elaborar un producto final que a futuro pueda ser comercializado o acoplado a sistemas de seguridad más robustos y con aplicaciones específicas.

1.3. Objetivos y alcance

El desarrollo de este proyecto incluye la implementación de una interfaz gráfica mediante una pantalla que incorpora la tecnología touch, el manejo de un módulo lector óptico de huellas digitales, la incorporación del sistema de acceso alternativo mediante la utilización de password y el accionamiento de una salida digital la cual a futuro podrá comandar actuador para cerradura eléctrica.

Se incluye además un sistema de monitoreo remoto al cual el usuario principal tiene acceso y mediante el cual se informa que usuario y a que hora tuvo acceso. No se incluye la opción de acceso remoto ni comunicaciones con otros sistemas.

La figura 1.2'' ilustra la disposición de hardware del sistema y la figura 1.3'' muestra la estructura en modulos funcionales del mismo.

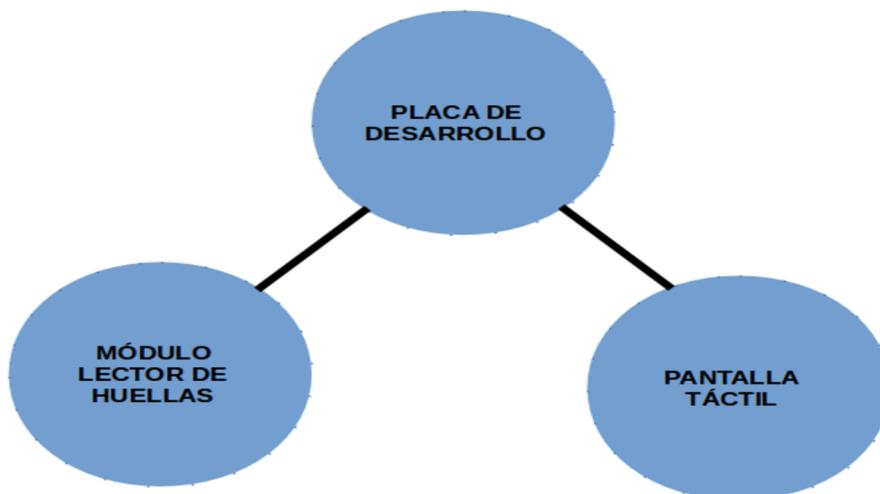


FIGURA 1.2: Disposición de hardware del sistema.

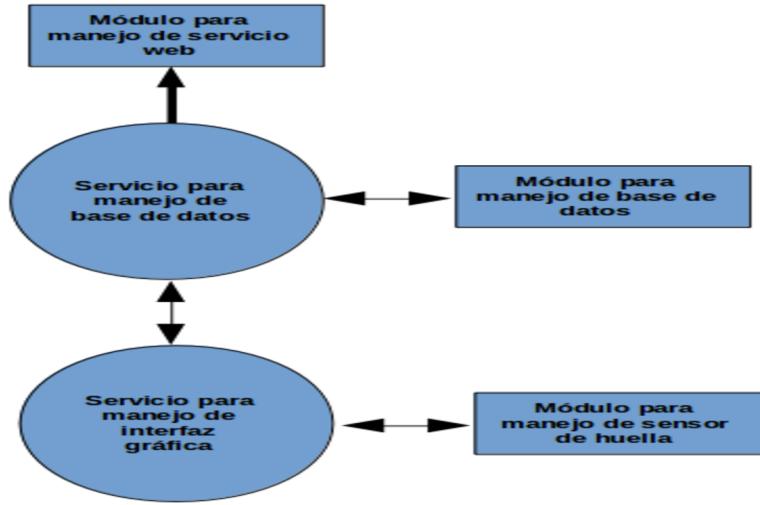


FIGURA 1.3: Módulos funcionales del sistema.

El proyecto esta basado en los siguientes lineamientos:

- Los Componentes para el desarrollo son abiertos y de libre acceso.
- Interfaz amigable con el usuario final.
- Mantiene escalabilidad para futuras implementaciones.

El proyecto desarrollado incluyó:

- Selección de la plataforma de desarrollo y hardware asociado.
- Selección del módulo lector de huella.
- Selección de la plataforma para implementación de interfaz gráfica.
- Selección de la plataforma para base de datos.
- Implementación de mecanismos de comunicación con el módulo lector de huellas
- Selección del sistema operativo embebido.
- Documentación del proceso de desarrollo.

El proyecto desarrollado no incluyó:

- Diseño del circuito impreso.
- Selección de unidades de alimentación principal.
- Mecanismos de seguridad referentes al manejo de información.

Capítulo 2

Introducción Específica

En esta sección se presenta el contexto en el cual se realizó el diseño, la descripción de los requerimientos y las fases mas importantes de la ejecución del proyecto.

2.1. Planteamiento

El sistema implementado se centra en la utilización de una placa de desarrollo embebida, el uso de un módulo-sensor lector de huellas, una pantalla para interacción con el usuario y el diseño y utilización de software necesario para la correlación de los diferentes elementos constitutivos. La placa de desarrollo embebida puede ser definida como un sistema electrónico basado en microprocesador diseñado para realizar funciones dedicadas. En el caso del módulo-sensor de adquisición de huella digital se trata de un dispositivo capaz de capturar la imagen de la huella, procesarla y almacenarla para su posterior utilización. Para una fácil manipulación del sistema por parte del usuario, se incorpora una interfaz gráfica desplegada sobre una pantalla táctil. En el caso del software, se selecciona como base una distribución del sistema operativo Linux para sistemas embebidos.

2.1.1. Descripción funcional

La única vía para acceder al sistema por parte del usuario es através de la interfaz gráfica la cual cuenta con un menú de tres partes las cuales son:

- Menú para acceso biométrico (acceso por reconocimiento de huella).
- Menú para acceso mediante validación de clave numérica personal.
- Menú para configuraciones.

De esta forma el usuario puede acceder únicamente si su huella o clave esta registrada con anterioridad. Cabe mencionar que la asignación de claves y registro de usuarios es privilegio del usuario principal. En caso de ser reconocida la huella o clave del usuario, se activa una salida digital de la placa de desarrollo (que en el futuro comandará una actuador final) y se despliega en pantalla un mensaje de notificación de bienvenida o caso contrario un mensaje de usuario no reconocido. El usuario principal tiene además la posibilidad de acceder a una pagina web donde se guarda el registro de los últimos accesos. La figura 2.1" muestra la secuencia funcional del sistema.

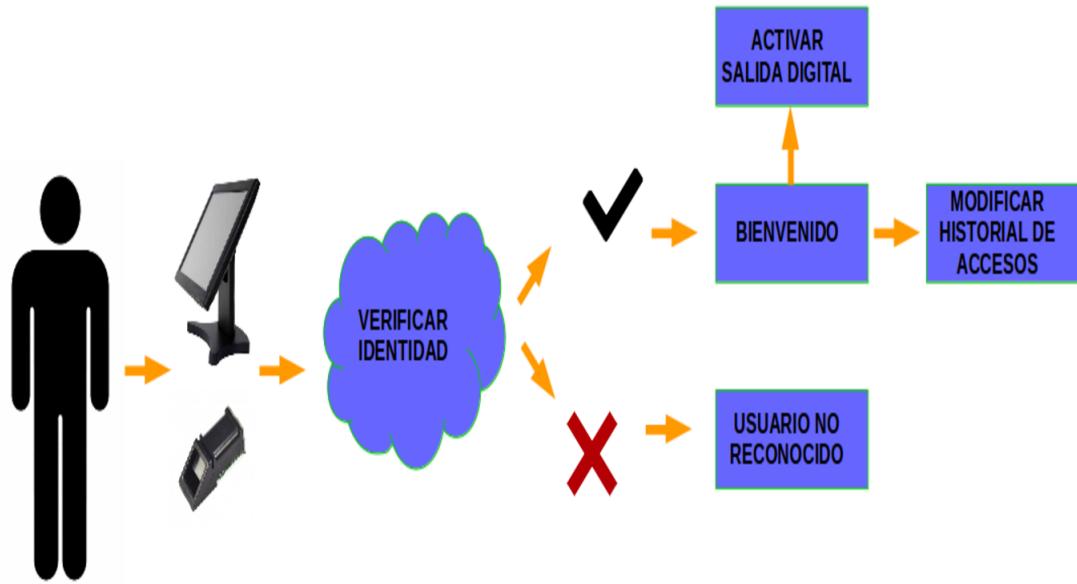


FIGURA 2.1: Secuencia funcional del sistema.

Para lograr este objetivo se diseñó un firmware con los siguientes módulos y sub módulos operativos.

- dbService: Servicio para manejo de base de datos y servidor TCP.
- logicService: Servicio para manejo de interfaz gráfica y cliente TCP.
- adafruit fingerprint: Sub módulo para manejo del sensor de huella.
- db lib: Sub módulo para lectura escritura de base de datos sobre SQLite.
- web: Sub módulo para desarrollo de la interfaz web.

Al desarrollar un sistema modular de este tipo se logra convertir la aplicación de un sensor de reconocimiento de huella en un posible sistema distribuido de seguridad mas completo ya que resulta relativamente sencillo anexar mas clientes TCP que pueden a su ves manejar nuevas interfaces gráficas con nuevos sensores y con distintas funcionalidades nuevas.

2.2. Requerimientos

A continuación el detalle de los requerimientos.

1.- Requerimientos del sistema:

- 1.1. Interfaz de usuario simple y claro.
- 1.2. El sistema posee dos modos de operación principales, biométrico y password.
- 1.3. En modo password se realiza la identificación del usuario mediante la validación de una clave numérica de cuatro dígitos.

- 1.4. En modo biométrico se realiza la identificación del usuario mediante la validación del patrón característico de la huella digital correspondiente al dedo indice de la mano derecha.
- 1.5. El sistema permite a un usuario identificado como principal, modificar la base de datos del sistema permitiéndole crear nuevos usuarios o borrarlos.
- 1.6. El usuario identificado como principal tiene la opción de acceder a un registro de actividad que le informe que usuario y a que hora tuvo acceso.
- 1.7. El sistema implementa el modo de acceso permitido como la activación temporal de una salida digital de la placa de desarrollo y un mensaje en pantalla.

2.- Componentes del sistema:

- 2.1 Se utiliza una placa de desarrollo embebida comercial para control central del proyecto.
- 2.2 El sistema cuenta con un módulo lector de huellas.
- 2.3 Forma parte del prototipo un módulo para la interfaz gráfica con tecnología touch.

2.3. Planificación

A continuación se muestra el desglose de tareas del proyecto. 1. Planificación (35h).

- Realizar plan del proyecto.(15h)
- Realizar el análisis de factibilidad.(10h)
- Realizar la gestión de calidad.(10h)

2. Investigación Preliminar.(105 hs)

- Buscar info. sobre sistemas de control de acceso por parámetros biométricos.(10 hs)
- Buscar información sobre módulos de adquisición de huellas dactilares.(10 hs)
- Buscar información sobre módulos display con tecnología touch.(10 hs)
- Buscar información acerca de entornos para programación de GUIs.(15 hs)
- Buscar información sobre bibliotecas para el uso de pantallas touch.(20 hs)
- Buscar info. sobre bibliotecas para módulos de adquisición de huellas dactilares.(20 hs)
- Buscar información sobre aplicaciones web para monitoreo remoto.(20 hs)

3. Selección de módulos comerciales y plataformas para el proyecto.(130 hs)

- Selección y pruebas preliminares del módulo y display touch.(30 hs)

- Selección y pruebas preliminares del módulo de adquisición de huellas dactilares.(30 hs)
 - Selección y pruebas preliminares del entorno de programación de GUIs.(45 hs)
 - Selección y pruebas para aplicación web para monitoreo remoto. (25hs)
4. Desarrollo de la interfaz de usuario (GUI).(40 hs) 5. Desarrollo de firmware.(200h)
- Implementar las funciones para la obtención de datos desde el módulo biométrico.(30 hs)
 - Implementar funciones para el desarrollo del módulo de monitoreo.
 - Corrección de errores.
6. Integración del sistema. (100h) 7. Procesos finales. (100h)

El diagrama Activity on Node final del proyecto se observa en la figura 2.2'', en esta se puede apreciar el orden seguido para la ejecución de las tareas, las cuales en total suman 710 horas.

Puede notarse también que el camino crítico el cual contempla las tareas que no pueden tener retraso (caso contrario comprometerían el tiempo de ejecución y entrega del proyecto) requieren una inversión de 535 horas.

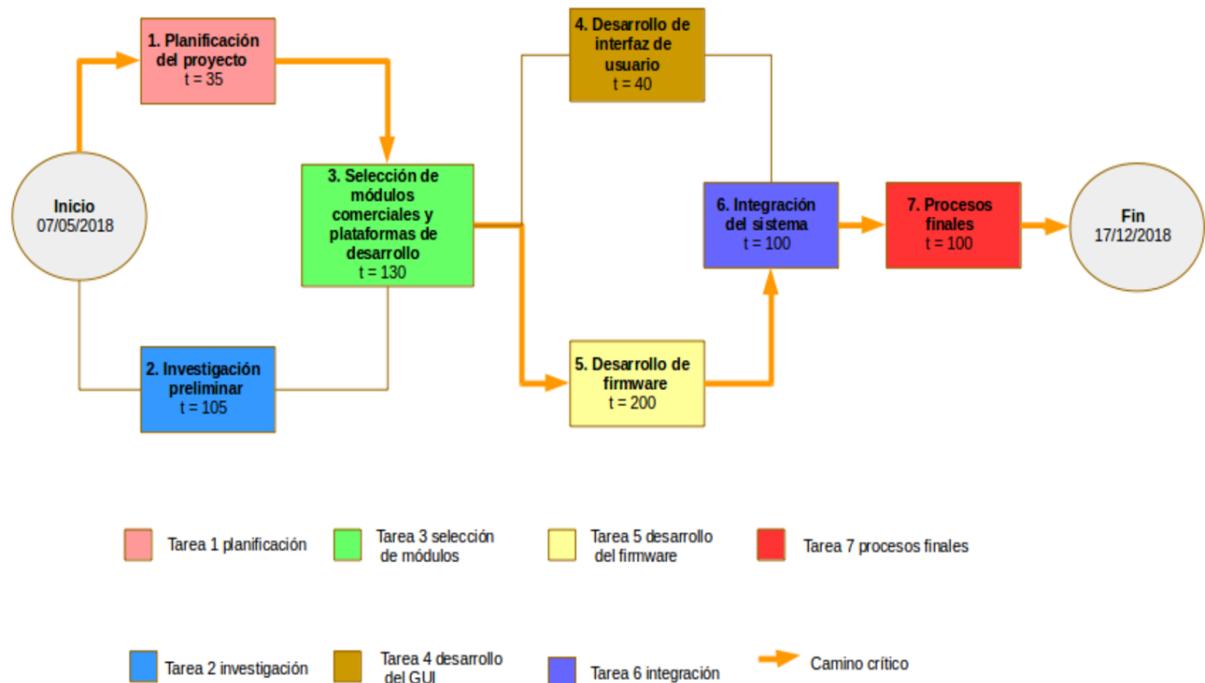


FIGURA 2.2: Diagrama Activity on Node.

Capítulo 3

Diseño e Implementación

3.1. Plataforma de desarrollo

Uno de los aspectos mas importantes dentro de la ejecución del proyecto es la correcta selección de la plataforma de desarrollo ya que de esta dependerá todas las funcionalidades a crear, además, una correcta selección puede facilitar la implementación de los diferentes requerimientos, es así que una mala decisión en este punto puede ocasionar retrasos significativos e incluso puede comprometer la culminación del proyecto. Para el caso puntual de esta aplicación al considerar que los requerimientos demandan alta capacidad de almacenamiento, soporte para sistema operativo, interfaces de comunicación, conectividad y uso de periféricos externos y sin dejar de lado la parte económica, se establece como la mejor opción la placa Raspberry Pi 3 modelo B+ cuyas principales características se detallan a continuación. Raspberry Pi es un ordenador de bajo costo pero altamente potente y de dimensiones muy pequeñas , fue desarrollada con la intención de facilitar la enseñanza de la informática y para que pueda ser utilizada para grandes proyectos como para aficionados de la electrónica y computación. Se han lanzando ya varios modelos previos de esta placa los mismos que incorporan nuevas características y funcionalidades sin que esto signifique incompatibilidad entre los diferentes modelos. Características técnicas:

- Procesador: Broadcom BCM2837BO Cortex-A53 (ARMv8) 64 bit
- Frecuencia: 1,4 GHz
- Memoria: 1 GB LPDDR2 SDRAM
- Conectividad inalámbrica: 2.4 GHz / 5 GHz IEEE 802.11b/g/n/ac Bluetooth 4.2 BLE
- Conectividad de red: Gigabit Ethernet over USB 2.0
- Puertos:
 - GPIO 40 pines
 - HDMI
 - 4xUSB 2.0
 - CSI (Cámara Raspberry)
 - DSI (Pantalla táctil)
 - Toma auriculares / video compuesto

- Micro SD
- Micro USB (Alimentación)

Otra de las ventajas del modelo seleccionado es el soporte wifi 802.11ac además de la opción de soporte para el estándar Power-over-Ethernet (PoE). La figura 3.1 muestra la distribución de los diferentes módulos constitutivos de la Raspberry.

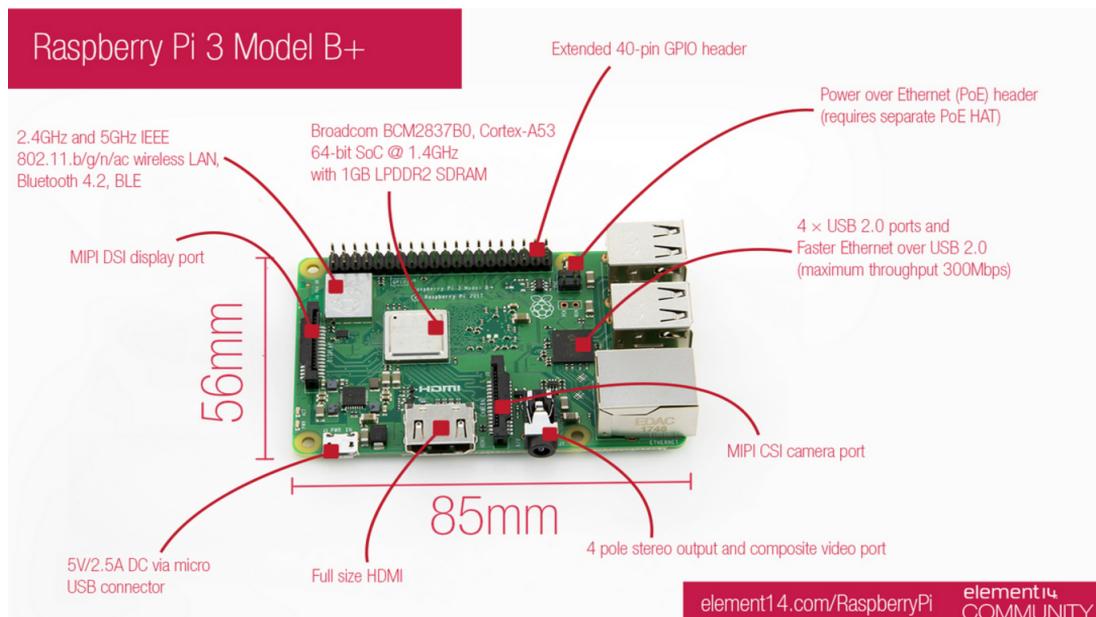


FIGURA 3.1: Raspberry Pi 3 B+ módulos constitutivos.

3.1.1. Linux embebido

El término Linux embebido se refiere al uso del núcleo Linux en un sistema embebido, en este caso la placa de desarrollo seleccionada. Dicho núcleo combinado con un conjunto de utilidades de software libre se ajustan dentro del hardware para el desarrollo del proyecto. Para tal propósito se ha seleccionado el sistema operativo Raspbian que es la distribución por excelencia para Raspberry, tal distribución es la mas completa y optimizada de las existentes la cual cuenta también con el respaldo oficial. Raspbian OS esta basada en la distro Debian Wheezy (Debian 7.0) optimizada para Raspberry. La figura 3.2 muestra la versión Raspbian Lite, versión con interfaz de escritorio, ejecutándose sobre la placa embebida.

3.1.2. Pantalla táctil element 14

Pantalla de 7 pulgadas diseñada especialmente para la placa de desarrollo seleccionada y gran variedad de placas de la misma familia. Este accesorio ha sido integrado en gran cantidad de proyectos tanto de información y entretenimiento así como de tecnología. Entre sus principales características se citan:

- Dimensiones : 194mm x 110mm x 20mm
- Resolución: 800 x 480 pixeles

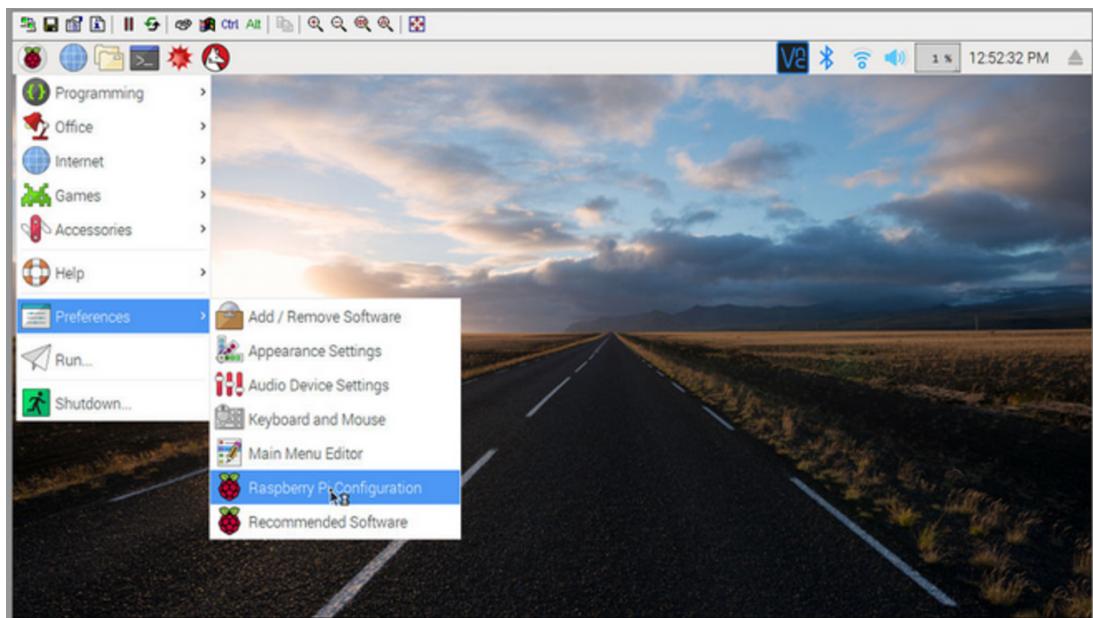


FIGURA 3.2: Interfaz Raspbian Lite.

- Tipo de detección: Capacitivo

Para su conexión con la Raspberry Pi utiliza una placa para adaptar señales de alimentación y señales lógicas, además, la comunicación se realiza mediante el puerto y protocolo DSI (Display Serial Interface). Gracias a la implementación del sistema operativo, la pantalla funciona como un dispositivo “plug and play” por lo que no fue necesario el desarrollo de librerías especiales. La figura 3.2 muestra los elementos constitutivos y necesarios para la conexión de la pantalla con la placa de desarrollo.

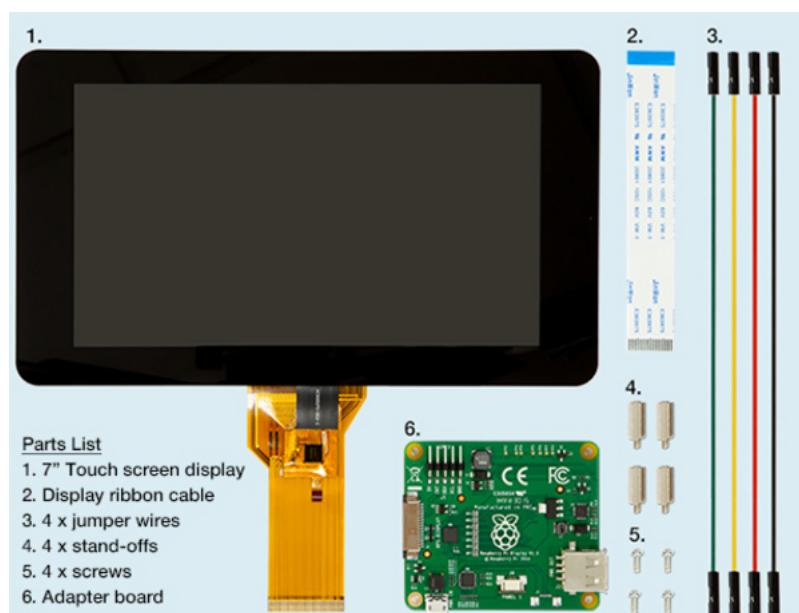


FIGURA 3.3: Pantalla touch 7 pulgadas element 14 y accesorios para conexión.

3.2. Desarrollo de interfaz gráfica

3.2.1. GTK+

GTK+ o GIMP Toolkit es una herramienta multiplataforma utilizada para crear interfaces gráficas de usuario, es una herramienta multiplataforma escrita en lenguaje C pero que permite el uso de otros lenguajes como Perl y Python.

GTK es software libre pero licenciado permite la creación de software libre como propietario. La herramienta se basa en una serie de librerías entre las cuales destacan:

- Glib: Proporciona los bloques básicos para construir aplicaciones y bibliotecas escritas en C, proporciona la implementación del bucle principal y funciones para el uso de cadenas y estructuras de datos comunes.
- Gobject: Proporciona el sistema para manejo de objetos para el diseño y dibujo de texto internacional.
- GIO: Proporciona una atracción del sistema de archivos permitiendo a las aplicaciones acceder a los archivos remotos y locales consistentemente.
- GTK: Proporciona controles de interfaz de usuario y señales para los controles de usuario.

3.2.2. Glade

Glade es un entorno gráfico para el desarrollo de interfaces gráficas basadas en GTK+ el cual genera un archivo en formato XML y gracias a esta propiedad permite el desarrollo con soporte para diferentes lenguajes de programación como C, Java, Python entre otros. La figura 3.4 muestra el entorno de desarrollo de Glade.

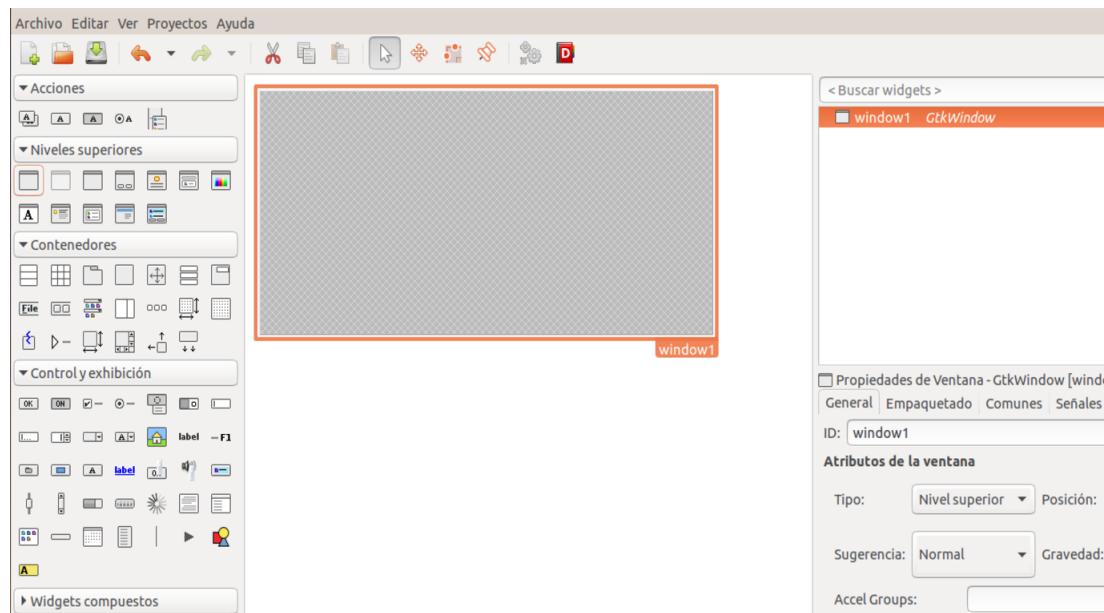


FIGURA 3.4: Entorno de desarrollo Glade.

El entorno se divide en tres columnas principales, a la izquierda la paleta de elementos con un conjunto de bloques como botones, contenedores y ventanas los cuales son posteriormente arrastrados hacia la columna central que ofrece la vista del proyecto. En esta locación se realiza la disposición de los componentes de nuestra interfaz según las necesidades del diseño.

Finalmente la columna de la derecha muestra el diagrama de árbol de la interfaz creada y muestra también las diferentes propiedades de cada elemento como sus dimensiones, etiqueta, nombre o identificador y señales asociadas que son funciones llamadas para su ejecución luego de ocurrido cierto evento ligado al mencionado elemento.

Para el desarrollo de la presente memoria se efectuó el siguiente método de trabajo para cada una de las páginas desarrolladas.

En primer lugar se selecciona una ventana principal, sobre esta se distribuyen el resto de componentes; para lograr una distribución mas uniforme y estructurada se hace uso de contenedores, una vez con los elementos en su posición final, se procede a establecer identificadores adecuados para cada uno así como la asignación de señales a eventos.

La asignación de señales e identificadores es muy importante ya que estos se utilizan luego en el programa principal. Finalmente todo el diseño se guarda como un archivo con extensión (.glade) el cual se invocará desde la aplicación final.

3.3. Módulo sensor para adquisición de huellas

3.3.1. Lectores ópticos reflexivos

El funcionamiento del sensor de huella comienza luego de colocar el dedo sobre la superficie de cristal del mismo la cual esta iluminada mediante un diodo led. La luz incide sobre el dedo causando el efecto de reflexión, este efecto se manifiesta con la aparición de zonas mas oscuras en las crestas de la huella y zonas mas claras en los valles.

La imagen recogida es procesada en busca de características relevantes (Minucias), luego de esto se genera una plantilla digitalizada la cual se guarda para ser consultada posteriormente. La figura 3.4 ilustra brevemente el proceso realizado por el sensor.

3.3.2. Módulo lector de huella Adafruit ZFM-20

Módulo basado en el chip AS608 para el procesamiento digital el cual permite la representación, el cálculo, extracción de características y cotejamiento de huellas. Entre los principales atributos se destacan:

- Fuente de alimentación: 3.6 – 6V.
- Corriente de operación: 120mA max.
- Tiempo de respuesta: <1s.
- Área activa del sensor: 14mm x 18mm.

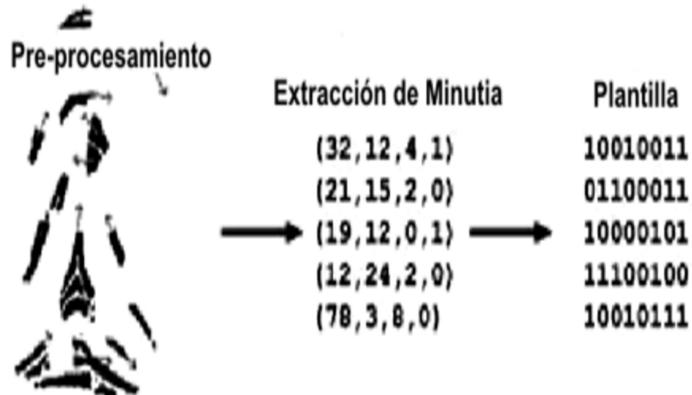


FIGURA 3.5: Procesamiento de huella dactilar.

- Capacidad de almacenamiento: hasta 162 patrones.
- Rangos de seguridad programables: 1 a 5.
- Taza de rechazo falso: <1
- Interfaz de comunicación: UART (TTL), USB.
- Velocidad de transmisión: 9600, 19200, 28800, 38400, 57600.

Otras características importantes de mencionar son la presencia de memoria RAM de 72K bytes, un buffer para almacenamiento de imágenes y dos buffers de archivo para procesamiento los cuales pueden ser accedidos aleatoriamente mediante las instrucciones del sistema.

El módulo posee además una base de datos conocida también como librería en la cual se realiza el almacenamiento de patrones de huellas en un espacio de memoria flash.

Para la comunicación del sistema con el sensor se utiliza el interfaz UART y para tal propósito el dispositivo lector de huellas establece su propio protocolo el cual consta de comandos y datos enviados en paquetes con los elementos mostrados en la tabla 3.1.

Existen diversos comandos que permiten trabajar con todas las funcionalidades del módulo lector, la tabla 3.2 muestra un resumen de los principales.

Una vez configurados los parámetros para la comunicación que por defecto están establecidos con un baud rate de 57600, 8 bits de datos y 1 bit de parada; se envía la trama iniciando desde el bit más significativo respetando el orden mostrado en la tabla 3.1. En el caso de la recepción de la respuesta el procedimiento es el mismo.

La figura 3.6 muestra el ejemplo del formato de envío del código para adquirir la imagen de una huella y el formato de respuesta esperado.

En el ejemplo expuesto, la confirmación por parte del sensor podría tener tres posibles casos listados a continuación:

- Código de confirmación = 0x00 correspondiente a un proceso exitoso.
- Código de confirmación = 0x01 correspondiente a una recepción de datos fallida.

TABLA 3.1: Elementos que conforman el paquete de datos para el protocolo de comunicación

Nombre	Símbolo	Tamaño	Descripción
Inicio	START	2 bytes	Valor fijo 0xEF01.
Dirección	ADDER	4 bytes	Permite la identificación de varios sensores sobre un mismo sistema, por defecto su valor es 0xFFFFFFFF.
Id de paquete	ID	1 byte	0x01 indica envío de comando, 0x07 indica respuesta del sensor.
Longitud	LENGTH	2 bytes	Tamaño de la carga de datos a recibir
Datos	DATA	–	Pueden ser instrucciones, datos o parámetros.
Checksum	SUM	2 bytes	Utilizado para comprobar un envío correcto.

TABLA 3.2: Principales comandos para el módulo lector de huella

Tipo	Código	Descripción
Sistema	0x12	Cambiar contraseña.
Sistema	0x15	Cambio de dirección.
Comunicación	0x0e	Cambio de parámetros.
Imagen	0x01	Cargar imagen.
Imagen	0x02	Extracción de característica.
Imagen	0x03	Montaje de imágenes.
Proceso	0x04	Buscar modelo.
Proceso	0x06	Gaurdar modelo.
Proceso	0x0c	Borrar modelo.

- Código de confirmación = 0x02 correspondiente a una adquisición vacía, es decir, no hay un dedo sobre el sensor.

3.4. Sistema General

3.5. Subsistema GUI (logic Service)

3.6. Subsistema para control del sensor de huella

3.7. Subsistema data-web

Formato de envío de datos.

Inicio	Dirección	ID	Longitud	Datos/Instrucción	checksum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xef01	0xffffffff	0x01	0x0003	0x01	0x005

Formato de recepción de datos

Inicio	Dirección	ID	Longitud	Datos/Confirmación	checksum
2 bytes	4 bytes	1 byte	2 bytes	1 byte	2 bytes
0xef01	0xffffffff	0x07	0x0003	--	0x005

FIGURA 3.6: Formato para envío/recepción de paquetes de datos.

Capítulo 4

Ensayos y Resultados

4.1. Pruebas funcionales del hardware

La idea de esta sección es explicar cómo se hicieron los ensayos, qué resultados se obtuvieron y analizarlos.

Capítulo 5

Conclusiones

5.1. Conclusiones generales

La idea de esta sección es resaltar cuáles son los principales aportes del trabajo realizado y cómo se podría continuar. Debe ser especialmente breve y concisa. Es buena idea usar un listado para enumerar los logros obtenidos.

5.2. Próximos pasos

Acá se indica cómo se podría continuar el trabajo más adelante.