



**UNIVERSIDAD
DE ANTIOQUIA**

**DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE
MONITOREO PARA
RE BALANCEO Y CONTROL DE EFICIENCIAS
(MARCE)**

Autor

Alejandro Jaramillo Mira

Universidad de Antioquia

**Facultad de Ingeniería, Departamento de ingeniería
electrónica y telecomunicaciones**

Medellín, Colombia

2020



DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE MONITOREO PARA
RE BALANCEO Y CONTROL DE EFICIENCIAS (MARCE)

Alejandro Jaramillo Mira

Informe de práctica o monografía o investigación o tesis o trabajo de grado
como requisito para optar al título de:
Ingeniería electrónica.

Asesores (a) o Director(a) o Co- Directores(a).

Pedro Pablo Botero Campuzano

Ingeniero electrónico

Álvaro López González

Ingeniero electrónico

Universidad de Antioquia
Facultad de Ingeniería, Departamento de ingeniería electrónica y telecomunicaciones.
Medellín, Colombia
2020.

TABLA DE CONTENIDO

Definiciones.....	6
Resumen.....	7
Introducción.....	7
Objetivos.....	8
General.....	8
Específicos.....	8
CAPÍTULO I.....	8
Marco Teórico.....	8
1.1 Python.....	9
1.1.1 Definición.....	9
1.1.2 Características.....	10
1.1.3 Ventajas y Desventajas.....	10
1.2 Comunicación UDP.....	10
1.2.1 UDP INICIALIZACIÓN.....	11
1.2.2 UDP WRITE.....	12
1.2.3 UDP LISTEN.....	12
1.3 Carpetas compartidas.....	12
1.3.1 Definición.....	12
1.3.2 Montaje de carpeta compartida.....	12
1.3.3 Características.....	13
1.3.4 Ventajas y desventajas.....	13
1.3.4.1 Ventajas.....	13
1.3.4.2 Desventajas.....	13
1.4 Raspberry pi	13

1.4.1 Definición.....	13
1.4.2 Características.....	14
1.4.3 Ventajas y Desventajas.....	15
1.5 Programación gráfica-GUI (PyQt- QtCreator).....	15
1.5.1 Definición.....	15
1.5.2 Características.....	16
1.5.3 Ventajas y desventajas.....	16
1.6 Modelo vista-controlador.....	16
1.6.1 Definición.....	16
1.6.2 Características.....	17
1.6.3 Ventajas y desventajas.....	18
CAPÍTULO II.....	18
Metodología.....	18
2.1 Familiarización con los procesos de la empresa.....	19
2.1.1 Proceso en cadena.....	19
2.2 Análisis de requisitos.....	20
2.3 Lectura de información de la plantilla.....	20
2.4 Autenticación.....	23
2.4.1 Ingreso de la terminal en su lugar físico.....	23
2.4.2 Matrícula de la operaria y su operación.....	24
2.5 Conteo de prendas y seguimiento de tiempos.....	24
2.6 Ingreso de inventario.....	24
2.7 Ingreso de activo.....	24
2.8 Paros en la producción.....	24
2.9 Cálculo de potencial y esperadas.....	24

2.10	Reporte al servidor central.....	24
2.11	Lectura al servidor del estado de la producción.....	25
2.12	Diseño de funcionamiento e interfaz gráfica.....	25
2.12.1	Pantalla inicial.....	25
2.12.2	Pantalla de matrícula física.....	25
2.12.3	Producción.....	26
2.12.4	Paros.....	27
2.12.5	Opciones.....	28
2.12.6	Inventario.....	28
2.12.7	Activo.....	29
2.12.8	Operario.....	29
2.12.9	Operación.....	30
2.13	Diseño e implementación de comunicación con el servidor.....	30
2.13.1	Estructura y tipos de eventos enviados al servidor.....	31
2.13.2	Vector de estados.....	34
2.13.3	Estructura de eventos enviados a la terminal.....	34
2.13.4	Lectura de información del servidor.....	35
2.14	Diseño e implementación de <i>login</i> , <i>logout</i> y paros del operario.....	35
2.14.1	<i>login and logout</i>	35
2.15	Paros.....	36
2.16	Implementación de cálculos y gestión de tiempos de las pulsadas.....	36
2.16.1	Pulsaciones.....	36
2.16.2	Esperadas por cuota y por potencial.....	36
2.16.3	Esperada por cuota.....	37
2.16.4	Esperadas potencial.....	37

2.16.5 Cálculo de potenciales por cuartiles.....	37
2.17 Pruebas de operatividad y funcionalidad.....	38
CAPÍTULO III.....	38
RESULTADOS.....	38
3.1 Dispositivo final.....	38
3.2 Interfaz de usuario.....	39
3.2.1 Pestaña inicial.....	39
3.2.2 Pantalla matrícula física.....	40
3.2.3 Pantalla producción.....	40
3.2.4 Paros.....	41
3.2.5 Activo e inventario.....	42
3.2.6 Operario (selección del nombre del operario).....	42
3.2.7 Operación (selección de la operación a desempeñar).....	43
3.3 Uso de la aplicación por parte de los supervisores y analistas.....	43
CONCLUSIONES.....	44
REFERENCIAS.....	45
INDICE DE IMAGENES	
Figura 1. Diagrama de operaciones.....	9
Figura 2. Importar librería socket.....	11
Figura 3. Inicialización de socket.....	11
Figura 4. Instrucción <i>bind</i>	11
Figura 5. Instrucción <i>sendto</i>	12
Figura 6. Instrucción <i>recv</i>	12
Figura 7. Instrucción montaje carpeta compartida.....	13
Figura 8. Raspberry pi 3 B+.....	14
Figura 9. Lógica modelo vista-controlador.....	17

Figura 10. Diagrama de procesos en cadena.....	20
Figura 11. Esquema de la plantilla en producción parte 1.....	21
Figura 12. Esquema de la plantilla en producción parte 2.....	23
Figura 13. Pantalla inicial de la aplicación.....	23
Figura 14. Pantalla de matrícula física de la aplicación.....	26
Figura 15. Pantalla de producción.....	27
Figura 16. Pantalla de paros.....	27
Figura 17. Pantalla de opciones.....	28
Figura 18. Pantalla de ingreso de inventario.....	28
Figura 19. Pantalla de ingreso de activo.....	29
Figura 20. Pantalla de matrícula de operario.....	29
Figura 21. Pantalla de matrícula de operación.....	30
Figura 22. Estructura del archivo de <i>logueo</i>	36
Figura 23. Fórmula esperadas por cuota.....	37
Figura 24. Fórmula esperadas por potencial.....	37
Figura 25. Gráfica de cuartiles.....	37
Figura 26. Fórmula de potencial.....	38
Figura 27. Pantalla final del dispositivo final.....	39
Figura 28. Pantalla final del inicio.....	40
Figura 29. Pantalla final matrícula física.....	40
Figura 30. Pantalla final de producción.....	41
Figura 31. Pantalla final de paros.....	41
Figura 32. Pantalla final de activo e inventario.....	42
Figura 33. Pantalla final de selección de operario.....	43
Figura 34. Pantalla final de selección de operación.....	43
Tabla 1. Cronograma de actividades.....	18
Tabla 2. Requisitos de la aplicación.....	20

Tabla 3. Puertos entre el servidor y el cliente.....	31
Tabla 4. Descripción del evento generado.....	32
Tabla 5. Descripción del evento recibido por la terminal.....	35

Definiciones

Canaleta: Ubicación física donde se va a confeccionar la referencia.

Referencia: Es un conjunto de prendas que se agrupan en un nombre y código.

Servidor: En este contexto, se define como la parte de la aplicación que guarda y gestiona las peticiones provenientes de los clientes, está desarrollado en *LabView*.

Terminal: Nombre que se le da al dispositivo compuesto por la aplicación y la Raspberry.

Asignación: Plantilla en Excel que tiene información relevante sobre la referencia a producir en una canaleta.

Operación: Acción que se ejecuta para completar la confección de la prenda.

Operario/a: Persona que realiza la operación.

Analista: Ingeniero encargado del módulo de aumentar la producción de la referencia que se está confeccionando en un módulo.

Supervisor: Persona líder de varios módulos, y superior a los analistas. Se encarga de asignar los módulos que van a confeccionar cada una de las referencias.

Módulo: Conjunto de personas que se agrupan para confeccionar una referencia.

Eficiencia: Capacidad de cumplir una meta.

Monitoreo: Control en el desarrollo de una acción.

Rebalanceo: Es la acción de los analistas para aumentar la producción. Esta puede ser, por ejemplo, asignar más personal, mover máquinas o cambiar la distribución de las operarias.

Restricción: En este contexto, se refiere a una operación que limita el flujo de la producción, haciendo que sea menor la cantidad de prendas confeccionadas por hora.

Auditoría militar: Proceso mediante el cual se valida, de forma rigurosa, la integridad de la prenda confeccionada.

DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE MONITOREO PARA REBALANCEO Y CONTROL DE EFICIENCIAS (MARCE).

Resumen

El trabajo realizado se centró en desarrollar e implementar una aplicación cliente de un sistema cliente-servidor. Esta aplicación está orientada al monitoreo del desempeño de los operarios en el proceso de confección de una prenda en su lugar de trabajo en la empresa *Crystal S.A.S.* recopilando datos y realizando cálculos con base en ellos que permitan sugerir, si es necesario, una nueva distribución del módulo tal que se logre aumentar la eficiencia de la producción final.

Esta aplicación además hace un seguimiento del inventario de cada una de las operaciones matriculadas en el módulo, lo que permite reportar en tiempo real, en su respectiva terminal, el cumplimiento o no de sus metas de producción.

Esta aplicación se construyó en paralelo con la aplicación cliente, que agrupa los datos provenientes de las terminales de cada módulo en un solo lugar y permite la visualización de los datos generados en una pantalla en cualquier dispositivo pc de la planta.

El resultado obtenido en este trabajo consiste en el diseño, desarrollo e implementación de la aplicación cliente desde su funcionamiento, interfaz gráfica y transmisión de datos al servidor central y que permite recopilar los datos de los operarios en tiempo real para su almacenamiento, y cálculo de algunos parámetros correspondientes a la operación que se ejecutará en el módulo en producción.

Introducción

Crystal S.A.S es una empresa de construcción y comercialización de marcas en Latinoamérica con más de 50 años de experiencia. Produce y comercializa las marcas *gef*, *punto blanco*, *baby fresh*, *galaxy* y *casino*. Los procesos productivos están integrados, desde el desarrollo de la materia prima, su conversión en productos de moda y su posterior ubicación en los puntos de venta, con procedimientos integrados de monitoreo y regulación en las distintas etapas de la confección para garantizar la mayor calidad en todos sus productos.

Actualmente las industrias quieren ser más competitivas, entrar en el mundo del *big data* y la industria 4G. En ese sentido *Crystal S.A.S* no se quiere quedar atrás, y tiene como una de sus metas seguir desarrollando tecnologías para mejorar la calidad y eficiencia en sus productos.

Por esta razón se desarrolla la aplicación llamada MARCE (sistema de Monitoreo Automático de Rebalanceo y Cálculo de Eficiencias) la cual recopila información generada por varios dispositivos colocados en cada módulo asignado a un operario, posteriormente la procesa en un servidor central para calcular las eficiencias y el cumplimiento de cada una de las

operaciones desarrolladas, y de esta manera poder atacar las restricciones del proceso y lograr aumentar la eficiencia del mismo.

Objetivos

General: Diseño y construcción de un sistema de recolección de datos y monitoreo de productividad en un proceso de manufactura textil en cadena aplicado al área de confección (MARCE).

Específicos:

El trabajo tiene como objetivos particulares los siguientes: Diseño e implementación de:

- Un sistema basado en *Raspberry pi* para la recepción y envío de información vía *WiFi* entre un sistema central y los módulos de producción locales con entradas análogas y con interfaz gráfica de usuario.
- Un algoritmo que integre la información de producción recibida desde el sistema central con los datos análogos y la interfaz gráfica de usuario del sistema local para la realización de cálculos de algunos parámetros de la producción
- Un algoritmo que realice cálculos relacionados con la productividad de un proceso que servirán en la toma de decisiones de producción.
- El protocolo de comunicación adecuado para el sistema, de acuerdo con sus necesidades técnicas y funcionales.
- Un sistema de *login* que permita la identificación de la persona que está trabajando y su respectiva operación asignada.
- Un sistema de conteo y monitoreo de tiempos en producción.

CAPÍTULO I

Marco Teórico

En este capítulo, se presentan los conceptos básicos, tanto en la parte técnica como en la de producción, necesarios para el desarrollo del trabajo propuesto y se realiza.

En la parte técnica se describen los conceptos manejados en ingeniería electrónica tales como desarrollo de software, programación de dispositivos embebidos, selección e implementación de protocolos de comunicación, captura y procesamiento de información y desarrollo de interfaces, entre otros.

La producción de una prenda en la empresa CRYSTAL S.A.S, contiene un flujo de trabajo denominado diagrama de operaciones (**Ver figura 1**), que define cada uno de los procesos necesarios para partiendo de la materia prima, se confecciona la prenda y luego pase al punto de venta.

En el diagrama de operaciones se resalta en amarillo el proceso hacia el cual se enfoca la aplicación en el área de producción, la que se encarga directamente de la confección y terminación de la prenda, para luego ser enviada a auditoría para su revisión de calidad y, si es necesario, un reproceso de la prenda.

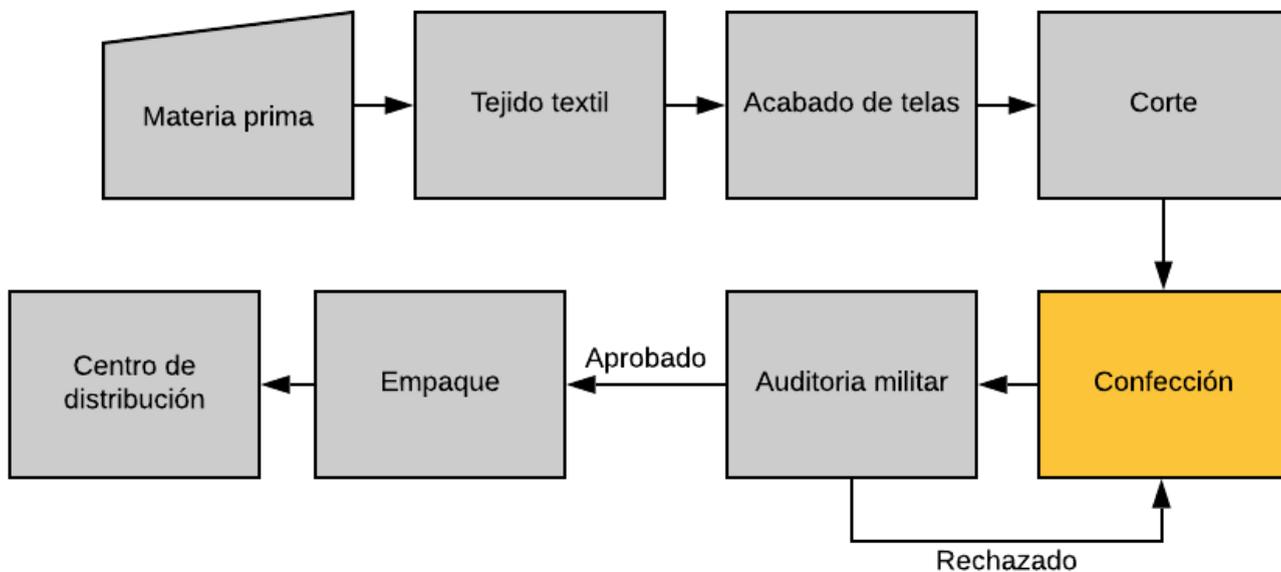


Figura 1. Diagrama de operaciones.

1.1 Python

1.1.1 Definición

Según la definición encontrada en wiki (2) es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, de tipado fuerte, dinámico y multiplataforma.

Python se basa en algunas filosofías que ayudan mucho en la solución de tareas difíciles. Algunos de los principios más conocidos son:

- Bello es mejor que feo.
- Explícito es mejor que implícito.
- Simple es mejor que complejo.
- Complejo es mejor que complicado.

1.1.2 Características

- Es de propósito general, no tiene un fin predeterminado.
- Es Multiparadigma. Aunque *Python* está orientado a objetos, se puede programar de distintas filosofías que se apliquen mejor al problema en particular o al estilo de programación.
- Es interpretado ya que no se requiere traducir el código fuente a código máquina, sino que se ejecuta directamente el código con ayuda de un intérprete.
- Es multiplataforma, Gracias a la característica de ser un lenguaje interpretado. Por esta razón se puede ejecutar en diferentes sistemas operativos (*Windows, Mac y Linux*), ya que existen intérpretes de código Python para estos.
- Tipado dinámico: Las variables que declaramos se adaptan automáticamente al tipo de dato necesario (*string, int, float*).
- Orientado a objetos, esta característica ayuda a la reutilización de código gracias a los conceptos de herencia y polimorfismo,
- Es de distribución libre y posee una licencia de código abierto llamada *Python Software Foundation License*. De esta forma mucha gente contribuye con librerías y nuevas funciones.

1.1.3 Ventajas y Desventajas

Al igual que otros lenguajes de programación, *Python* tiene ventajas y desventajas, entre las cuales se enumeran las siguientes:

Ventajas

- Simplificado, ordenado y rápido: Al ser un lenguaje de alto nivel y, muy cercano al lenguaje humano, es muy rápido de programar y, además, su sintaxis ayuda a llevar un orden en el código para una mejor interpretación.
- Abundancia de bibliotecas: Al ser un lenguaje de programación tan popular, la comunidad desarrolla bibliotecas especializadas para temas en específico de cada área.
- Gracias a que la comunidad se mantiene desarrollando nuevas librerías para *Python*, el soporte que se puede encontrar es muy variado, y de mucha utilidad al momento de empezar a trabajar con librerías específicas, o encontrar la solución a muchos errores.

Desventajas

- Curva de aprendizaje en web es compleja: Python es un lenguaje que no fue orientado para ese tipo de desarrollo y por lo tanto se presentan obstáculos al momento de construir funciones que de otra forma serían más fáciles de implementar en lenguajes para esta plataforma.

- Más lento que sus competidores por la interpretación que debe hacer el sistema para poder correr el código sin ningún problema
- Conexión con servidores difícil: La mayoría de servidores no tienen soporte para *Python*, y si lo tienen es lento y complicado de implementar.

1.2 Comunicación UDP:

Python tiene librerías diseñadas para lograr la comunicación entre equipos por medio del protocolo *UDP (User Datagram Protocol)*.

A diferencia de *TCP (Transmission Control Protocol)*, que garantiza la entrega ordenada de la información a su destinatario asegurando primero una conexión entre los dispositivos, *UDP* no establece una comunicación previa, no asegura la llegada de ella, ni tampoco su orden ya que los paquetes enviados pueden adelantarse unos a otros.

En sistemas donde el envío de información debe ser rápido, el sistema *UDP* es de gran ayuda, porque tiene cabeceras simples, lo que no provoca una carga significativa adicional en la red.

Para utilizar el protocolo *UDP* con *Python* se dispone de la librería **socket** la cual ayuda en la configuración y envío de la información por medio de una *IP* y un puerto de conexión. Se invoca de la siguiente manera (**Ver figura 2**):

```
import socket
```

Figura 2. Importar librería *socket*.

INICIALIZACIÓN de UDP

La instrucción de la **figura 3** define la inicialización de la variable *socket*. La primera constante define el tipo de dirección. La segunda el tipo de protocolo, en este caso *UDP*.

```
socket_pulsacion = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
```

Figura 3. Inicialización de *socket*.

La instrucción *bind* que aparece en la **figura 4**, se usa para la inicialización de la escucha de *UDP*, recibe los valores de la *IP* del dispositivo, y el puerto donde escucha las peticiones.

```
CONN2 = (ip_terminal, puerto_pulsacion_rx)
socket_pulsacion.bind(CONN2)
```

Figura 4. Instrucción *bind*.

UDP WRITE

La instrucción `sendto` de la **figura 5**, envía una cadena de letras (*string*) en forma de *bytes* a la IP seleccionada por el puerto especificado.

```
CONN6 = (ip_servidor, puerto_pulsacion_tx)
socket_pulsacion_tx.sendto(bytes(respuesta), CONN6)
```

Figura 5. Instrucción `sendto`.

UDP LISTEN

La instrucción `recv` de la **figura 6**, recibe el tamaño del buffer de la trama, para regular el tamaño de la información. Está la guarda para ser usada.

```
ready= select.select([socket_multicast],[],[],0.018)
if ready[0]:
    trama=socket_multicast.recv(131072)
```

Figura 6. Instrucción `recv`.

1.3 Carpetas compartidas

La comunicación con el servidor se dividió en dos partes para no saturar de manera significativa los canales disponibles. Para ello se decidió que se leyeran los datos que no se actualizan frecuentemente a través de una carpeta compartida que es visible para los demás dispositivos en la red interna de la empresa. La información restante, se transmite por medio de eventos generados entre los dispositivos.

1.3.1 Definición

Las carpetas compartidas proporcionan un lugar de almacenamiento común, en una red local o interna, llamada intranet, que se pueden ver y editar con los permisos de usuario correctos y proporcionan un sistema centralizado de información en un sólo punto, de fácil acceso para los dispositivos de la red.

1.3.2 Montaje de carpeta compartida

La carpeta compartida tiene una dirección local que será la ruta donde se van a ver todos los archivos, y la dirección del dispositivo que se quiere montar. En la **figura 7** se muestra la sintaxis de inicialización en un sistema basado en *Linux*.

```
mount [OPTION...] DEVICE_NAME DIRECTORY
```

Figura 7. Instrucción montaje carpeta compartida.

1.3.3 Características

- Permiten acceder desde otros equipos, por medio de la red local, a los archivos compartidos
- Gestión de permisos de lectura y escritura para distintos usuarios
- Si varios usuarios trabajan simultáneamente en un archivo, pueden visualizar los cambios guardados.

1.3.4 Ventajas y desventajas

Ventajas

- Distintos usuarios pueden trabajar sobre la misma información.
- Otros usuarios pueden ver la información remotamente.
- Se centralizan los datos.

Desventajas

- No se puede editar simultáneamente el mismo archivo.
- Otros usuarios pueden ver nuestra información.

1.4 RASPBERRY PI

Se escogió para este proyecto en específico una *Raspberry pi 3 B+* (Ver **figura 8**) debido a su gran versatilidad, su pequeño tamaño y a su pantalla *touch*, lo que hace que sea una gran herramienta en el puesto de trabajo de la operaria. Una ventaja adicional es la de permitir ejecutar programas basados en Linux, lo que ayuda a la optimización de procesos por su bajo consumo de recursos.

1.4.1 Definición

Es un ordenador de placa reducida o de placa simple (SBC) de bajo coste desarrollado en el reino unido por *Raspberry Pi Foundation* con el objetivo de estimular la enseñanza de la informática en las escuelas.

Este dispositivo no incluye periféricos externos como por ejemplo teclado o ratón, aunque para su adaptación se han desarrollado algunos, tales como pantalla *touch* y cámaras.

Se volvió una gran herramienta fuera de su mercado objetivo, siendo muy utilizado en aplicaciones comerciales, privadas, y de entretenimiento.

La fundación *Raspberry Pi Foundation* tiene contratos con dos empresas para su distribución y venta y aunque es un producto con propiedad registrada sobre el que mantiene control, deja su uso libre tanto de manera particular como educativo.

El software es de código abierto y su sistema operativo oficial basado en *Linux* llamado *Raspbian*, es una versión adaptada de *Debian*, aunque permite usar otros tales como una versión de *Windows 10*.

El sistema está desarrollado principalmente para el aprendizaje de *Python*, aunque soporta otros lenguajes tales como *C*, *Ruby* y *Perl*.



Figura 8. Raspberry pi 3B+

https://es.m.wikipedia.org/wiki/Archivo:Raspberry_Pi_4_Model_B_-_Side.jpg

1.4.2 Características

La *Raspberry pi 3 B+* al ser un ordenador de placa reducida, tiene especificaciones iguales a un computador de escritorio o portátil, pero más reducidas en capacidad por su tamaño. El dispositivo tiene las siguientes características:

- CPU + GPU: *Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz*
- RAM: *1GB LPDDR2 SDRAM*
- WiFi + Bluetooth: *2.4GHz y 5GHz IEEE 802.11.b/g/n/ac, Bluetooth*
- *4.2, BLE*
- Ethernet: *Gigabit Ethernet sobre USB 2.0 (300 Mbps)*
- GPIO de 40 pines
- HDMI

- 4 puertos *USB 2.0*
- Puerto *CSI* y *DSI* para conectar una cámara y una pantalla táctil
- Salida de audio estéreo y vídeo compuesto
- *Micro-SD*
- *Power-over-Ethernet (PoE)*

Además de estas características físicas, el sistema operativo está diseñado para trabajar principalmente con el lenguaje de programación *Python*.

1.4.3 Ventajas y Desventajas

Ventajas:

- Sistema operativo basado en *Linux* de software libre.
- Bajo consumo de energía.
- Comunidad grande con varios proyectos desarrollados.
- Tamaño reducido.
- Precio bajo alrededor de unos \$35 USD.

Desventajas:

- Memoria y procesador reducidos.
- No cuenta con sistema de refrigeración propio.
- En varios foros de internet, incluyendo el oficial, la gente reporta muchos problemas con la comunicación vía *WiFi*, tales como desconexión de la red sin previo aviso y lentitud en la conexión.

1.5 Programación gráfica-GUI (*PyQt - Qtcreator*)

En el proyecto se utiliza la aplicación *Qtcreator* junto con la librería *PyQt* para diseñar la interfaz gráfica que es el puente entre la programación y el usuario.

La librería *PyQt* consume pocos recursos, y es usada para crear interfaces de usuario simples que no necesitan grandes funciones.

1.5.1 Definición

Por su definición en *Wikipedia GUI (graphical user interface)* es un programa informático que actúa de interfaz de usuario que utiliza tanto un conjunto de imágenes y objetos gráficos para representar la información como acciones disponibles en la interfaz. Su objetivo es proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

1.5.2 Características

- Elementos gráficos para realizar acciones intuitivas.
- Hace más sencillo el manejo de una aplicación.
- Fácil curva de aprendizaje

1.5.3 Ventajas y Desventajas

Ventajas:

La GUI permite:

- Una fácil interacción con el usuario.
- El despliegue de diálogos para guiar al usuario en el uso del programa.
- La ejecución de múltiples programas de forma simultánea.

Desventaja:

- Son costosos computacionalmente.

1.6 Modelo vista-controlador

La aplicación desarrollada en este proyecto se hizo con base en el modelo vista-controlador (Ver **figura 9**), el cual divide las funciones en tres bloques principales que se encargan de tareas específicas desde la interacción externa, hasta el procesamiento de la información y su visualización para presentarla al usuario final.

Este modelo de programación permite la programación eficiente de la aplicación final, porque se pueden dividir los problemas mayores en tareas específicas que son más fáciles de realizar.

1.6.1 Definición

EL modelo vista-controlador (MVC) separa la lógica de programación en tres componentes principales, modelo, vista, y controlador, los cuales tienen asignadas tareas específicas dentro de la aplicación.

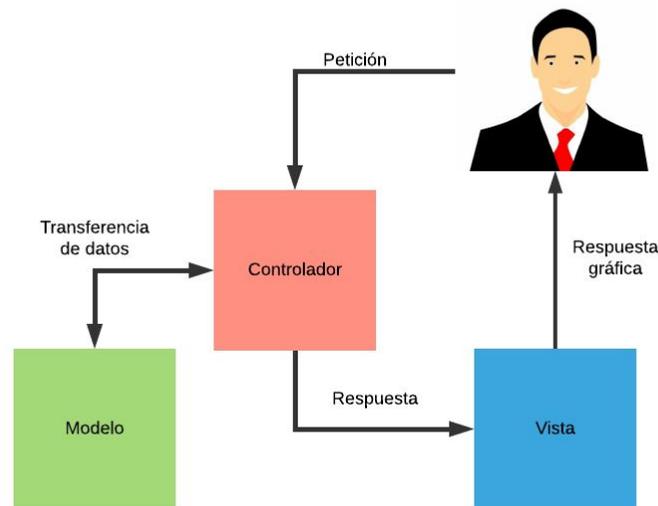


Figura 9. Lógica modelo vista-controlador.

El Modelo gestiona todos los accesos a la información, así como las peticiones que llegan desde el controlador; Además se encarga de las consultas a las bases de datos, de la actualización y de las búsquedas que luego son retornadas o mostradas en la vista.

La Vista realiza la representación visual de la información entregada desde el modelo de una manera organizada para el usuario, además, se encarga de mostrar las opciones que el controlador puede desarrollar y gestionar.

El Controlador es el puente entre la vista y el modelo. Se encarga de recibir y gestionar todos los eventos de entrada que se generan en la aplicación, principalmente por el usuario, como por ejemplo, presionar un botón en la interfaz o ingresar en un cuadro de texto alguna información.

El flujo general del modelo vista controlador se puede resumir en los siguientes puntos:

1. El usuario interactúa con la interfaz realizando alguna petición (puede ser presionar un botón, o ingresar información faltante).
2. El controlador recibe la petición, la gestiona y toma las decisiones apropiadas para resolver ese evento.
3. El controlador accede al modelo y actualiza lo necesario para cumplir con el evento generado por el usuario.
4. El controlador envía a la vista la información resultante de la actualización del modelo. Esta se organiza para mostrar al usuario el resultado del evento en la interfaz gráfica.

1.6.2 Características del modelo vista-controlador

- Distribución y organización en la programación.

- Reutilización de código.
- Mantenimiento más sencillo de la aplicación.
- Lógicas separadas de **vista** gestión y **controlador**.

1.6.3 Ventajas y desventajas

Ventajas:

- Es posible construir más vistas sin tener que modificar el modelo o el controlador.
- Al dividir en eventos separados las peticiones del usuario, se hace más fácil el manejo de estas.
- Al estar dividido en módulos, se facilita la corrección de errores.
- Implementación de forma modular.
- Fácil mantenimiento.

Desventajas:

- Curva de aprendizaje elevada
- La cantidad de archivos a manejar aumenta considerablemente.
- La separación en distintos bloques genera una gran complejidad en el sistema a desarrollar.
- Tiempo excesivo en la programación.

CAPÍTULO II

Metodología

La metodología definida para el desarrollo de la aplicación, fue dividirlo en ocho tareas principales (**Ver tabla 1**) que se describen a continuación, con un cronograma establecido con base en no solamente el tiempo necesario para el entendimiento y el levantamiento de los requisitos, sino también el periodo de duración de la práctica, la cual se extendió a dos semestres para asegurar la implementación correcta en producción. Adicionalmente en varios de estos procesos se tuvo en cuenta la espera necesaria para permitir que el otro equipo terminara el servidor, que se hacía de manera paralela para la finalización total.

Tabla 1. Cronograma de actividades.

Mes/Actividad	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6
Familiarización con los procesos de la empresa	X	X				

Entendimiento del problema y levantamiento de requisitos		X				
Diseño del funcionamiento e interfaz gráfica para el usuario.		X	X			
Diseño e implementación de comunicación con el servidor			X	X		
Diseño e implementación de <i>login</i> , <i>logout</i> y paros del operario				X	X	
Implementación de cálculos y gestión de tiempos de las pulsadas					X	
Pruebas de funcionalidad			X		X	X
Implementación en las plantas de producción						X

2.1 Familiarización con los procesos de la empresa

Esta actividad se desarrolló en dos etapas, la primera fue dedicada a conocer la parte del proceso de la prenda en la empresa en general, desde la materia prima, transformación, producción, y distribución hasta el punto de venta y, además el acople a las nuevas tareas y funciones que se debían cumplir en el tiempo de la práctica.

La segunda etapa se enfocó en la parte de producción de la prenda, hacia donde iba a estar dirigido el proyecto, para conocer el desarrollo de la misma área y las distintas personas involucradas. El quehacer estuvo más orientado a estudiar y justificar los requerimientos de la aplicación para su correcto planteamiento y desarrollo.

Partiendo inicialmente de los conceptos de producción y teoría de restricciones, se definen los requerimientos del área y los datos que deben tenerse en cuenta para la realización de los cálculos, necesarios en la implementación de la aplicación.

2.1.1 Proceso en cadena

Se denomina procesos en cadena (Ver **figura 10**) a la construcción y ensamble de las piezas de un artículo mientras que este va pasando por puestos de trabajos que tienen una tarea específica. En el caso de la empresa *Crystal*, la prenda en proceso se mueve por las máquinas que tienen tareas asignadas, denominadas operaciones y que son llevadas a cabo por personas llamadas operarios.

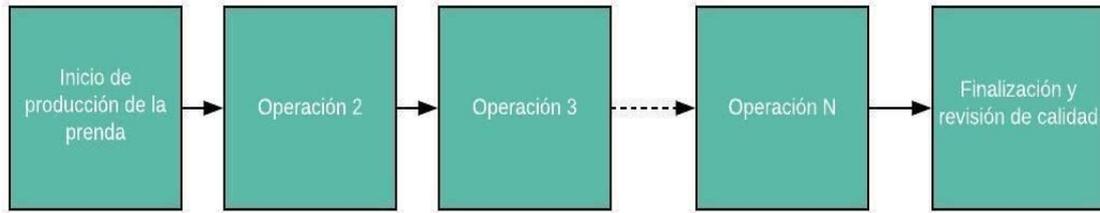


Figura 10. Diagrama de procesos en cadena.

De este tipo de proceso salen algunos parámetros a los cuales se les hizo monitoreo en la aplicación:

Cuota: meta en unidades que se deben producir en un lapso de tiempo definido, usualmente una hora.

Potencial: Es lo que el operario es capaz de producir en un intervalo de tiempo, definido en este proyecto como una hora.

Inventario: Cantidad de unidades para procesar que tiene el operario acumuladas. Este inventario depende directamente de la operación previa, que es la encargada de surtir a la siguiente.

En estos conceptos se basó la construcción y desarrollo de la aplicación, ya que ellos indican a los analistas y supervisores si es necesario modificar la distribución de las operaciones o de los mismos operarios para una mayor producción.

2.2 Análisis de requisitos

Tras analizar y estudiar el proceso , hacer seguimiento a varios módulos y tener reuniones con el equipo de producción, se llegó finalmente a la tabla de requisitos principales (ver **tabla 2**) que debería llevar la aplicación final, enfocados principalmente en la aplicación cliente desarrollada en este informe.

Tabla 2. Requisitos de la aplicación.

Requisit o	Nombre del requisito
1	Lectura de información de la plantilla
2	Autenticación operario y operación
3	Conteo de prendas y seguimiento de tiempos
4	Ingreso de inventario

5	Ingreso de Activo
6	Paros en la producción
7	Cálculo de potencial y esperadas
8	Reporte al servidor central
9	Lectura al servidor del estado de la producción

2.3 Lectura de información de la plantilla

La plantilla (Ver **figura 11**) es un documento de *Excel* que contiene indicaciones al sistema de cómo se va a distribuir el trabajo en la canaleta correspondiente.

Este documento fue desarrollado por el área de producción para tener un control de los datos y distribución del módulo de confección.

La plantilla contiene los siguientes datos importantes que son leídos desde la aplicación para cumplir los requerimientos pedidos:

Planta	0									
Tipo Asignación	Primaria									
Código Referencia	632513									
Canaleta	20									
Módulo	2071									
Compromiso	2									
Número de personas	12									
Minutos Turno	570									
Factor Estándar	0.8									
Nombre Referencia	UNIB80677MM V NECK TEE									
Estándar consultado	11.802									
Estándar Tablero	86.544									
Costo	83									
Orden Op	Operación	C. Operación	Es real	Es Tablero	Prec 1	Prec2	Terminal	Estacion	Operaria	Cod. Operaria
1	UNIR HOMBROS	PEHOFIL1311S107	0.484	0.3872				2	ALVARO LOPEZ	43579791
2	ASENTAR HOMBROS	ASHOCLL1311S108	0.491	0.3928	1			2	DORIS LUCIA VALLEJO FORONDA	43425073
2	ASENTAR HOMBROS	ASHOCLL1311S108	0.491	0.3928	1			2	LINA MARIA MARTINEZ JARAMILLO	43833314
2	ASENTAR HOMBROS	ASHOCLL1311S108	0.491	0.3928	1			4	CLAUDIA VIVIANA TANGARIFE	43926078
3	CERRAR CUELLO V	CVCUPLA1111S109	0.4	0.32				1	ELVIRA LOPEZ DE FRIA	40917493
4	FIJAR CUELLO CONTOR	FVCUFLC2112S110	0.556	0.4448	3			3	ELVIRA LOPEZ DE FRIA	40917493
5	PEGAR CUELLO V	PVCUFLC2112S112	0.797	0.6376	4	2		6	YANETH PATRICIA LOZANO CASTANEDA	43206280

Figura 11. Esquema de la plantilla en producción parte 1

Planta: indica a qué planta física corresponde la plantilla "0" Sabaneta, "1" Pereira, "2" Manizales, "3" Armenia.

Tipo asignación: Indica qué importancia tiene la referencia con respecto a la prioridad del pedido, "Primaria" y "Secundaria".

Código de referencia: Es un número de seis dígitos que representa a la referencia que va a producción, se utiliza para hacerle seguimiento por todo el proceso de confección.

Canaleta: Número que corresponde al espacio donde se produce la referencia en cuestión. En este lugar van a estar los operarios y las máquinas organizadas para la elaboración correcta de la prenda final.

Módulo: Número de identificación que tiene un grupo de personas que confeccionan una referencia.

Número de personas: Cantidad de operarios que van a estar en la canaleta produciendo la prenda.

Minutos turno: Cantidad de tiempo al día que se trabaja en la producción de la prenda.

Factor Estándar: Porcentaje en que va a ser afectado el estándar consultado para obtener el estándar tablero, lo que influye directamente en la cuota final.

Nombre Referencia: Nombre código que se le da a la prenda para su seguimiento en las distintas etapas de su producción.

Estándar consultado: Cantidad de minutos que toma confeccionar una de las prendas de la referencia. Este tiempo es determinado previamente por un equipo de preparación.

Estándar Tablero: Cantidad de tiempo afectado por el factor estándar a partir del estándar consultado. Este número es en definitiva la cantidad de minutos en el que se debe confeccionar una prenda.

Cuota: Cantidad esperada de prendas confeccionadas en una hora. Esto se calcula con base en los minutos del estándar tablero.

Orden Op: Indica el número en el orden que corresponde a la operación.

Operación: Nombre de la operación a desarrollar.

C Operación: Código único de cada una de las operaciones necesarias para confeccionar una prenda.

Es real: Cantidad de minutos que se demora hacer la operación en la actual referencia (Estándar real).

Es Tablero: Cantidad de minutos afectados por el factor estándar a partir del estándar real.

Prec1: Precedente número 1 de la operación que necesariamente se debe hacer para poder hacer ejecutar la actual.

Prec2: Precedente número 2 de la operación que necesariamente se debe hacer para poder hacer ejecutar la actual.

Terminal: Número asignado a cada uno de los dispositivos para diferenciarlos y ubicarlos en la canaleta.

Estación: Número que indica la máquina en que se está trabajando. La mayoría de veces concuerda con el de la terminal.

Operaria: Nombre de la persona que va a realizar la operación.

Cod_Operario: Código que identifica a la persona en la base de datos de los empleados de la empresa.

Los siguientes parámetros están en la **figura 12**.

Cuota	¿Final?	Pos_Máq	Ocu_Maq	Per/Oper	Asignacion
83				0.671104	0.671104
83				0.68081	0.68081
83				0.68081	0.68081
83				0.68081	0.68081
83				0.554631	0.554631
83				0.770937	0.770937
75					1
83				0.919301	0.919301
83				0.880000	0.880000

Figura 12. Esquema de la plantilla en producción parte 2

Cuota: Cantidad de prendas que debe confeccionar la operación actual en una hora.

¿Final?: Indica con una "x" si la operación finaliza la producción de la prenda.

Per/Oper: Cantidad de personas necesarias por hora para poder confeccionar la prenda en esa respectiva operación.

Asignación: Cantidad de personas asignadas por hora a la operación actual.

2.4 Autenticación

Luego de obtener la información correspondiente, se realiza la autenticación, que es un proceso que se divide en dos partes, la primera es la matrícula de la terminal en su lugar físico, y segundo el ingreso de la operaria con su respectiva operación.

2.4.1 Ingreso de la terminal en su lugar físico

La terminal debe pasar por su matrícula física para descargar la información correspondiente a la referencia que se va a producir. En esta se debe seleccionar la planta, la canaleta y la terminal correspondiente.

2.4.2 Matrícula del operario y su operación

Luego de la matrícula física de la terminal, el operario busca y selecciona su nombre y la operación que está desarrollando. Así se identifica y se hace un seguimiento de su producción. Cualquier operario que esté en la plantilla de la respectiva combinación planta, canaleta, se puede matricular en cualquier terminal.

2.5 Conteo de prendas y seguimiento de tiempos

Se implementaron varios pulsadores en el dispositivo para que la operaria pudiera marcar la finalización de la construcción de una prenda. Con estos se puede contar la producción del día y, además, determinar cuánto tiempo se demora realizando la operación asignada. Estos dispositivos son la entrada principal de datos del sistema, ya que sin la señal de finalización de confección de prenda, no se puede tener un seguimiento preciso de la producción.

2.6 Ingreso de inventario

Este se debe ingresar debido a que la producción de una prenda puede haber empezado antes de implementar el sistema o a que el día anterior pudieron quedar unidades sin confeccionar, así que para hacer un correcto seguimiento del flujo de confección, se debe poder ingresar el inventario acumulado que tiene la operaria para desarrollar.

2.7 Ingreso de activo

Cada una de las máquinas utilizadas en la planta de confección tiene un número único de identificación.

2.8 Paros en la producción

Informa al sistema que hay un problema en producción el cual no deja que la operaria siga normalmente con su tarea. Este requisito monitorea el tiempo que se está en paro, además de poder escoger la razón principal de este, de una lista ya predefinida, por parte del equipo de producción.

2.9 Cálculo de potencial y esperadas

Satisface la principal necesidad de los analistas de conseguir la información de la operación en tiempo real, extrayéndose de los cálculos basados en la plantilla y la pulsación de la operaria cada vez que termina de confeccionar una prenda.

2.10 Reporte al servidor central

Se envían los cálculos, las matrículas, pulsaciones y paros al servidor central, el cual guarda la información para un análisis y visualización más detallado.

2.11 Lectura al servidor del estado de la producción

El servidor central genera una información grupal, como por ejemplo las unidades totales o las esperadas totales del módulo, que también debe ser mostrada en la terminal.

2.12 Diseño de funcionamiento e interfaz gráfica

Habiendo obtenido los requisitos técnicos del sistema, se empieza con el diseño de la interfaz gráfica sobre la plataforma *Raspberry*. Como se indicó en el capítulo 1 se usó *Qtcreator* para crear la maqueta de la interfaz gráfica y luego por medio de *Python* se añaden las funcionalidades correspondientes a cada una de las pestañas de la aplicación.

La interfaz gráfica se dividió en distintas tareas, que cumplen unas funciones específicas en la aplicación, basadas en el análisis de requisitos. A continuación se muestran cada una de las pestañas creadas y su respectiva función.

2.12.1 Pantalla inicial

Esta pestaña (ver **figura 13**) es la primera que se muestra al iniciar la aplicación. Se muestra la información de matrícula de la última vez que se usó. Si esta corresponde con los datos actuales, se presiona sí para seguir a producción iniciando el conteo de tiempo, si se presiona no, va a la misma pestaña, pero sin la sesión iniciada.

M.A.R.C.E. v 5.0	
Planta	88
Canaleta	88
Terminal	88
Activo	88
Referencia	88
Operario	88
Operacion	88

SI NO

Si la informacion corresponde a sus datos presione SI, en caso contrario, presione NO para matricularse.

Figura 13. Pantalla inicial de la aplicación.

2.12.2 Pantalla de matrícula física

Esta pestaña (**ver figura 14**) contiene por un lado los campos planta, canaleta y terminal que permiten modificar la matrícula física de la terminal. Luego de seleccionar lo necesario, se confirma con el botón modificar, pero si no se desea cambiar la información, se regresa con el botón Ir a producción. Además está el botón técnico que envía a opciones. Por otro lado contiene información sobre el nombre de la red, y la IP de la terminal.

M.A.R.C.E. v 5.0		
Planta	1	
Canaleta	5	
Terminal	2	
Red Actual	red	
IP	88.88.88.88	

Modificar

Ir a Produccion

Tecnico

Figura 14 .Pantalla de matrícula física de la aplicación.

2.12.3 Producción

Esta es la pestaña principal de la aplicación (**ver figura 15**), la cual muestra la mayoría de información relacionada con la producción y los datos importantes que visualizan los analistas o los supervisores, como por ejemplo las unidades o el potencial. Desde aquí se puede ingresar a las demás pestañas de matrícula.

Está dividida en 3 columnas principales. La primera se llama módulo, que contiene la información correspondiente a la matrícula física. Primero tiene el número correspondiente matriculado, luego continua el cumplimiento total, abajo de este, aparecen las unidades y las esperadas de la canaleta en ese instante de tiempo, la hora y un botón que nos guía a la pestaña de matrícula física, además de mostrar abajo la canaleta y la terminal correspondiente, por último tenemos el botón de activo y los botones de video y foto, que son para un próximo desarrollo.

La segunda columna contiene la información correspondiente al operario y operación, comenzamos con el cumplimiento, unidades y esperadas, luego el tiempo total que lleva la persona en producción y por último, un botón auxiliar para contar, si el pulsador externo falla en algún momento.

La tercera columna contiene en primer lugar el nombre y el código de la referencia, luego tenemos los dos botones de selección de operario y operación, luego tenemos el campo asociado al inventario actual, el campo de la derecha indica el inventario actual, y el de la izquierda además de mostrar el inventario inicial ingresado, es un botón que redirige a la pestaña

donde se ingresa el inventario. Luego tenemos el botón de iniciar sesión que inicia la matricula del operario, y el de paros que envía a la pestaña correspondiente. Tenemos un reloj que muestra el tiempo parcial que lleva el operario en producción, este tiempo tienen un color distinto si la operaria está matriculada (verde) y si no lo está (rojo). Por último tenemos los dos últimos botones de finalizar montaje y opciones, que para la parte de desarrollo no se conectaron y se dejaron disponibles para un futuro desarrollo.



Figura 15. Pantalla de producción.

2.12.4 Paros

Esta pestaña (ver **figura 16**) tiene las opciones para poder empezar y finalizar un paro correctamente. En el desplegable se tienen varias opciones como administrativo, ingeniería, mecánico, calidad e insumos, que fueron seleccionados por el área de producción. Luego tenemos una lista de daños que se deben seleccionar por motivos mecánicos. En medio de la pestaña tenemos el botón de regresar a producción, y el botón de iniciar el cronómetro. Si se quiere finalizar, es necesario escoger una opción de la lista de las causas seleccionables. Además de tener en la parte de abajo el contador del tiempo que lleva la terminal en este estado, y el botón asociado a la finalización del mismo.



Figura 16. Pantalla de paros.

2.12.5 Opciones

Esta pestaña (ver **Figura 17**) contiene las opciones para cerrar la aplicación. Contiene los botones apagar, reiniciar y el reset general que vuelve todos los valores a los de defecto.



Figura 17. Pantalla de opciones.

2.12.6 Inventario

En esta pestaña (ver **Figura 18**) se ingresa el inventario inicial de la operación actual matriculada. Tiene un teclado numérico, además de un botón de confirmar y regresar, aquí también se puede ver el inventario actual y el anterior ingresado.



Figura 18. Pantalla de ingreso de inventario.

2.12.7 Activo

En esta pantalla (ver **Figura 19**) encontramos el lugar donde se debe ingresar el activo correspondiente a la máquina que se está realizando la operación matriculada, similar a la pestaña de inventario, contiene un teclado numérico simple y, además, el activo anterior ingresado junto con un botón regresar para cancelar los cambios no confirmados.



Figura 19. Pantalla de ingreso de activo.

2.12.8 Operario

En esta pestaña (ver **Figura 20**) se encuentran los nombres de cada uno de los operarios que estaban en la plantilla luego de obtenerla después de la matrícula física de la terminal. A la izquierda cada uno de los nombres está en un botón, al seleccionarlo se mostrará en la barra de

la derecha el nombre, el cual se puede confirmar o de lo contrario volver a la pestaña de producción.

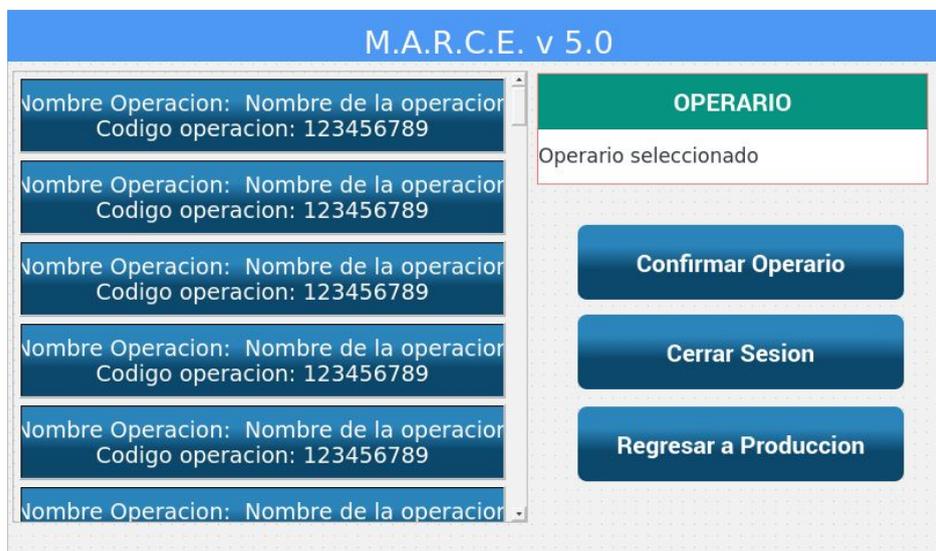


Figura 20. Pantalla de matrícula de operario.

2.12.9 Operación

En la **figura 21** se observa que esta pestaña tiene la misma estructura que la de operario. A la derecha podemos encontrar los botones de confirmación y regresar a producción. En la parte izquierda están los botones con cada una de las operaciones que se descargaron de la plantilla en la matrícula física.

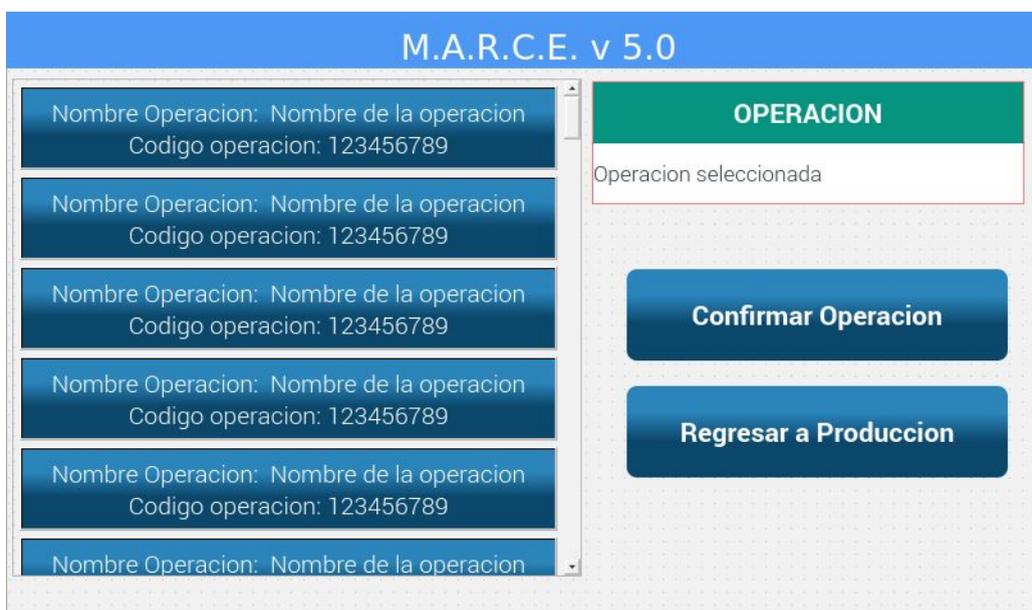


Figura 21. Pantalla de matrícula de operación.

2.13 Diseño e implementación de comunicación con el servidor

La comunicación debe realizarse de forma bilateral de tal manera que la terminal pueda

informar al servidor sobre los eventos ocurridos, y este pueda comunicar a aquella los cambios en las canaletas y en la producción, así como las unidades y la eficiencia de toda la referencia.

La comunicación se dividió en dos partes, una manejada por eventos, y la otra se basa en la lectura de carpetas compartidas. En la sección de *Python* del marco teórico (1.3) se introdujo la manera en la cual se comunica la terminal con el servidor utilizando UDP por unos puertos específicos que se muestran en la **tabla 3**:

Tabla 3. Puertos entre el servidor y el cliente.

	Recepción(Escucha)	Transmisión
Cliente	puerto pulsación = 65225 puerto matricula = 65221 puerto eventos = 65222 puerto vector_est = 65223 puerto multicast = 65224	puerto pulsación = 65270 puerto matricula = 65271 puerto eventos = 65272 puerto vector_est = 65273 puerto multicast = 65274 IP Servidor= 172.18.171.150
Servidor	puerto pulsación = 65270 puerto matricula = 65271 puerto eventos = 65272 puerto vector_est = 65273 puerto multicast = 65274	puerto pulsación = 65225 puerto matricula = 65221 puerto eventos = 65222 puerto multicast = 65224 IP Terminal = Dada en el evento

Cada evento generado en la terminal, se envía por un puerto distinto hacia el servidor. Así se asegura que los que ocurren con más frecuencia (pulsaciones) no tengan pérdidas ni apilamientos en la comunicación.

El puerto pulsación es donde se escuchan y envían los eventos que indican que hubo una pulsación por parte de la operaria.

El puerto matrícula es donde se envían y reciben los eventos que indican una nueva matrícula del operario en una terminal de trabajo.

El puerto eventos: es donde se envían y reciben los eventos asociados a inventario, activo, paros y cierre de sesiones.

El puerto vector_est: Allí se envía desde la terminal el vector de estados.

El puerto multicast: Es un puerto de escucha general para todas las terminales y también sirve para enviar un comando global.

2.13.1 Estructura y tipos de eventos enviados al servidor

Los eventos generados en la terminal tienen una estructura predefinida para asegurar una correcta llegada de la información al servidor. Para esto se creó una cadena de caracteres, con campos separados por ';', que se muestran en la **Tabla 4** con su nombre y descripción:

Tabla 4. Descripción del evento generado.

Comando	Descripción
Inicio Trama	Carácter 'T' que se comprueba en el servidor para asegurar una trama correcta.
Planta	Número de planta matriculado '0' Sabaneta, '1' Pereira , '2' Manizales, '3' Armenia.
Canaleta	Número de canaleta matriculada desde el '0' al '53'.
Terminal	Número de la terminal física matriculado desde el '0' al número '20'.
IP	Dirección IP de la terminal.
Length	Número de caracteres de la trama total.
Comando	Informa el estado de la terminal 'E' Producción
Tipo de evento	Combinación de caracteres que define qué tipo de evento se generó: 'P' pulsación, 'MA' Cambio de activo, 'MORARIO' Matrícula de operario, 'MOCION' Matrícula de operación, 'IP' Inicio paro, 'FP' Final de paro, 'CS' Cierre de sesión, 'I' Ingreso de inventario, 'REST' Evento de reseteo de la terminal.
Consecutivo del evento	Número que indica cuántos eventos se han generado en la terminal.
RTC del evento	Número en formato RTC que representa la hora en la que se generó el evento.
Fecha	Fecha de creación del evento.
Código de referencia	Número que identifica la referencia que se está confeccionando.
Código de operación	Identifica la operación única que está matriculada cuando se genera el evento.
Código de operaria	Número de identificación de la operaria que está matriculada en el momento.
Referencia	Nombre de la referencia que está en producción.
Operación	Nombre asociado a la operación que se está desempeñando.
Operaria	Nombre de la operaria que está trabajando en el momento.
Módulo	Número que identifica el grupo de personas que está trabajando en la referencia actual.

Inventario digitado	Inventario inicial ingresado por la operaria en la respectiva operación.
Cód. paro	Número que identifica el paro realizado en la terminal desde una lista predefinida.
Paro	Nombre del paro seleccionado.
Ubicación	Parámetro que identifica en qué zona de la producción se encuentra la terminal: 'Inicio' Las dos primeras terminales, 'Medio' todas las terminales hasta las 3 últimas, 'Final' las 3 últimas terminales.
Daño	Código asociado a un daño reportado desde la máquina cuando se genera un paro.
Nombre de daño	Nombre del daño reportado en la terminal.
Causa	Código de la causa por la cual se generó el paro o se reportó el daño.
Nombre de la causa	Nombre de la causa por la cual se generó un paro en la terminal
Ocupación de la máquina	Campo que indica cuántas personas están usando la máquina en la cual se encuentra la terminal.
Número de la operación	Número que se encuentra en la columna orden de la plantilla para la operación matriculada.
Talla medida (no usado)	Campo separado para futuro desarrollo
Nuevo estado	Estado de la terminal: ''Producción ',' Paro ',' Cierre de sesión'
Código de medida (no usado)	Campo separado para futuro desarrollo.
Código de aprobación montaje (no usado)	Campo separado para futuro desarrollo.
Nombre Medida (No usado)	Campo separado para futuro desarrollo.
Valor medida	Campo separado para futuro desarrollo.
Pulsadas OP-OP	Número de pulsadas que ha hecho la operaria en la combinación operario-operación actual

Ocupación de operario	Relación Tiempo a trabajar vs hora asignado al operario para trabajar en la operación matriculada.
Estándar operación	Estándar extraído de la plantilla asociado a la operación matriculada
Cuota operación	Número dado en unidades por hora que se deben producir en esa operación.
Tiempo asignación	Tiempo que toma en ejecutarse la operación.
ID terminal	Número que identifica a la terminal a nivel empresarial.
Código precedente 1	Código asociado a la operación que se realiza antes de ejecutar la operación matriculada.
Nombre de precedente 1	Nombre de la operación asociado a la operación que se realiza antes.
Código precedente 2	Código asociado a una segunda operación que se debe hacer antes.
Nombre de precedente 2	Nombre asociado a una segunda operación que se debe hacer antes.
Estado	Número que indica si la terminal está en producción '2', o paro '1'
RTC Creación del registro OP-OP	Tiempo en formato RTC que indica la primera matrícula del operario en la operación actual.

Observación en el consecutivo del evento.

Esta estructura se genera a partir de eventos específicos en la terminal, los cuales indican un cambio significativo que puede afectar los cálculos o la producción de la referencia.

Forma como se generan algunos eventos:

'P': cada vez que la operaria hace una pulsación en la combinación operario-operación correspondiente.

'MA': cuando se ingresa el activo de una máquina a la terminal.

'MORARIO': cuando el operario inicia sesión.

'MOCION': cuando se escoge la operación a trabajar.

'IP': cuando se inicia un paro de producción en la terminal.

'FP': cuando finaliza el paro que interrumpe la producción de esa operación.

'CS': cuando el operario termina de trabajar, normalmente al final del día.

'I': cuando se ingresa el inventario inicial asociado a la operación matriculada.

'REST': Cada que se resetean los valores por defecto de la terminal, reporta el eliminado de toda la información.

2.13.2 Vector de estados

Para asegurar la continua transmisión en tiempo real, la terminal envía al servidor su estado actual cada 5 segundos, el cual contiene un formato similar al enviado en los eventos, con la diferencia en el orden de los campos y su continua transmisión como ya se explicó. El servidor recolecta y actualiza la información en tiempo real de este vector de estados sobre lo ocurrido en las terminales, como los cálculos y el tiempo transcurrido.

2.13.3 Estructura de eventos enviados a la terminal

Luego de recibir el evento proveniente de la terminal, el servidor le envía una respuesta de confirmación para asegurar la entrega de todos los eventos que tiene represados. Se comprueba si el comando corresponde con sus datos matriculados y si es así, procede a la ejecución de una tarea específica (**Ver tabla 5**).

Tabla 5. Descripción del evento recibido por la terminal.

Comando	Descripción
Inicio trama	Letra 'S' Puesta al inicio de la trama para corroborar que la trama se generó correctamente
Planta	Número planta a la cual va dirigida el evento
Canaleta	Número de la canaleta a la cual se va a enviar el comando
Terminal	Número de la terminal que debe recibir el comando
Comando	Letra que identifica el comando a ejecutar en la terminal: 'H': Borrar evento, 'R' reiniciar aplicación.
Información comando	Información extra enviada por el servidor a la terminal, solamente si es necesario.

2.13.4 Lectura de información del servidor

La lectura al servidor se hace por medio de carpetas compartidas, que contienen la información necesaria para el funcionamiento correcto de la terminal. Tal información se reparte en tres: La plantilla, que contiene la información inicial para empezar la producción, el archivo de eficiencias, que contiene la eficiencia y potenciales de las canaletas matriculadas, y por último el archivo de matrículas, que indica los operarios se han matriculado, y en qué lugar de la planta.

2.14 Diseño e implementación de *login*, *logout* y paros del operario

2.14.1 *login* y *logout*

El sistema de *logueo*(Ver **Figura 22**) hace el seguimiento a las personas matriculadas en la terminal revisando en las carpetas compartidas que tiene con el servidor si alguna de las personas que se han matriculado antes generaron un evento de *login* en otro lugar, ya que una persona solamente puede estar matriculada en una sola terminal.

2	05_04_2019	634974	COD319	111111119	1	11	9	3637315110
2	05_04_2019	634976	COD321	131111121	1	13	11	3637309657
2	05_04_2019	634976	COD323	131111123	1	13	13	3637309685
2	05_04_2019	634974	COD321	111111121	1	11	11	3637315024
2	05_04_2019	634976	COD312	131111112	1	13	2	3637309705
2	05_04_2019	634974	COD314	111111114	1	11	4	3637315044
2	05_04_2019	634974	COD312	111111112	1	11	2	3637315083
2	05_04_2019	634976	COD318	131111118	1	13	8	3637309817
2	05_04_2019	634976	COD320	131111120	1	13	10	3637309851
2	05_04_2019	634976	COD322	131111122	1	13	12	3637309891
2	05_04_2019	634976	COD324	131111124	1	13	14	3637309932
2	05_04_2019	634977	COD315	141111115	1	14	5	3637309959
2	05_04_2019	634978	COD311	151111111	1	15	1	3637310078
2	05_04_2019	634977	COD319	141111119	1	14	9	3637310053
2	05_04_2019	634978	COD311	151111129	1	15	1	3637310072
2	05_04_2019	634978	COD313	151111113	1	15	3	3637310103
2	05_04_2019	634978	COD317	151111129	1	15	7	3637310154
2	05_04_2019	634977	COD318	141111118	1	14	8	3637310154
2	05_04_2019	634978	COD313	151111130	1	15	3	3637310094
2	05_04_2019	634978	COD317	151111117	1	15	7	3637310162
2	05_04_2019	634978	COD319	151111130	1	15	9	3637310191
2	05_04_2019	634978	COD319	151111119	1	15	9	3637310205
2	05_04_2019	634978	COD323	151111123	1	15	13	3637310249
2	05_04_2019	634979	COD317	161111117	1	16	7	3637310247
2	05_04_2019	634979	COD319	161111119	1	16	9	3637310262
2	05_04_2019	634979	COD321	161111121	1	16	11	3637310274

Figura 22. Estructura del archivo de *logueo*.

El archivo tiene la información correspondiente a cada uno de los operarios matriculados en la aplicación en el día. Como se puede ver en la **figura 22**, los campos relevantes, para poder saber si el operario se *logueo* en otro lugar, son *los campos* de código operario, tiempo en formato RTC, planta, canaleta y la terminal.

Para generar un *login*, la operaria debe matricularse en la terminal y este evento ser enviado al servidor, el cual lo guarda en el archivo de *login*, para que quede disponible para todas las terminales y así puedan revisar si deben cerrar la sesión de alguien en producción.

2.15 Paros

Los paros son una parte muy importante del manejo de tiempos, ya que no deben contar en la producción, pero se deben tener en cuenta para analizar, posteriormente, en qué momentos y por qué razones se pierde tiempo en producción. Este tiempo sigue corriendo en paralelo con el tiempo matriculado del operario, así que luego, al finalizar la jornada, se restan los tiempos de producción y de paro, para no afectar negativamente el cumplimiento o el potencial del operario.

2.16 Implementación de cálculos y gestión de tiempos de las pulsadas

2.16.1 Pulsaciones

Las pulsaciones generan eventos que indican el tiempo en el cual se hizo. Al momento de contar, se guarda en un archivo distinto la información del intervalo entre la última y la actual hecha que será usada en los cálculos de las esperadas, el cumplimiento y el potencial.

2.16.2 Esperadas por cuota y por potencial

Las unidades esperadas son las que se espera que estén producidas en ese instante de tiempo.

2.16.3 Esperada por cuota

Las esperadas por cuota se calculan (ver **Figura 23**) con el valor de la cuota de la plantilla, este valor está en unidades/hora.

$$\textit{Esperadas cuota} = \textit{Cuota} * (\textit{Tiempo total op} - \textit{op})$$

Figura 23. Fórmula de cálculo de las esperadas por cuota.

2.16.4 Esperadas potencial

Se calculan (**Figura 24**) obteniendo cuántas unidades debería sacar la operaria en una hora, esto se hace con ayuda de los cuartiles.

$$\textit{Esperadas por potencial} = \textit{potencial} * (\textit{Tiempo total op} - \textit{op})$$

Figura 24. Fórmula para el cálculo de esperadas por potencial.

2.1.6.5 Cálculo de potenciales por cuartiles

Las pulsaciones generadas por la operaria describen una función de probabilidad normal (Ver **figura 25**). Gracias a esto se puede dividir en cuatro partes la distribución de la variable aleatoria con respecto a la cantidad de datos, a esta división se le llama cuartil. Cuando se divide en **cuatro partes de 0% -25%, 25%-50%, 50%-75% y 75%, 100%**, reciben el nombre de cuartiles. Al dividir nuestros datos en cuatro partes, podemos eliminar los dos sectores de **0%-25%** y **75%-100%** de los datos, así sólo nos quedamos con los datos significativos de la distribución.

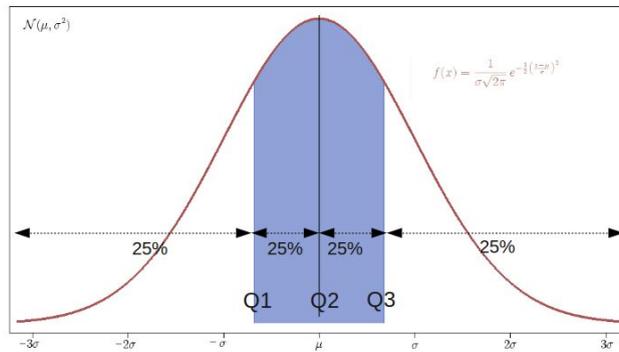


Figura 25.Gráfica de cuartiles.

Teniendo los datos del centro de la distribución, podemos sacar un promedio de ellos, y así hallar el tiempo que la operaría se demora haciendo una prenda, esto es lo que se conoce como potencial (ver **figura 26**). El resultado se da en unidades/tiempo.

$$Potencial = (\sum Rango\ Intercuartil) \div número\ de\ datos$$

Figura 26.Fórmula de potencial.

2.17 Pruebas de operatividad y funcionalidad

En esta etapa del desarrollo, se realizan las pruebas técnicas en busca de fallos y errores que puedan afectar la funcionalidad del sistema. En el próximo capítulo se muestran los resultados finales de la implementación de la aplicación y se hace un análisis de estos.

CAPÍTULO III

RESULTADOS

Se presentan los resultados obtenidos luego del desarrollo e implementación del sistema en la planta, incluido un comentario del uso del dispositivo por parte del área de producción.

3.1 Dispositivo final

Como se planteó en el marco teórico, el hardware utilizado para este proyecto es una *Raspberry pi 3*. En la **figura 27**, se muestra el dispositivo funcionando en una máquina de la planta.

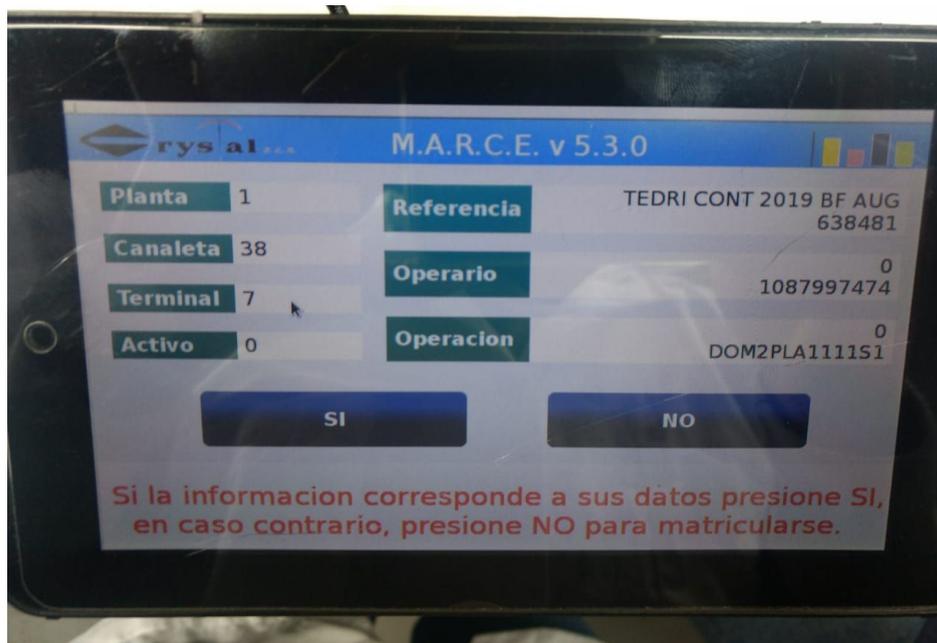


Figura 27. Pantalla del dispositivo final.

3.2 Interfaz de usuario

Es una de las partes fundamentales del desarrollo, ya que el proceso de la matrícula y uso deben ser sencillos para cualquier persona. En la metodología se explicó el planteamiento de diseño para la interfaz gráfica, y el funcionamiento de cada una de las pestañas que componen la aplicación, cuya función se enumera a continuación.

- Pestaña inicial (Matrícula directa o continúa a producción).
- Pantalla matrícula física (Matricula física de la terminal).
- Producción (Pantalla principal de la aplicación).
- Paros (Inicio y finalización)
- Opciones (Reinicio y apagado de la terminal)
- Inventario (Ingreso del inventario inicial a la aplicación)
- Activo(Ingreso del activo de la máquina en la aplicación)
- Operario (Selección del nombre de la operaria)
- Operación (Selección de la operación a desempeñar)

3.2.1 Pestaña inicial

En esta sección se muestra la interfaz de pestaña inicial (Ver **figura 28**), que, como su nombre lo indica, es la primera ventana que el usuario se encuentra al iniciar el dispositivo. Para hacer mucho más rápida la matrícula, se muestran los datos que antes estaban matriculados en la terminal y si son los correctos, el usuario tiene la posibilidad de oprimir el botón 'SI' y continuar

directamente con la etapa de producción, si no lo son puede presionar el botón 'NO' para proceder con la matrícula de la terminal.

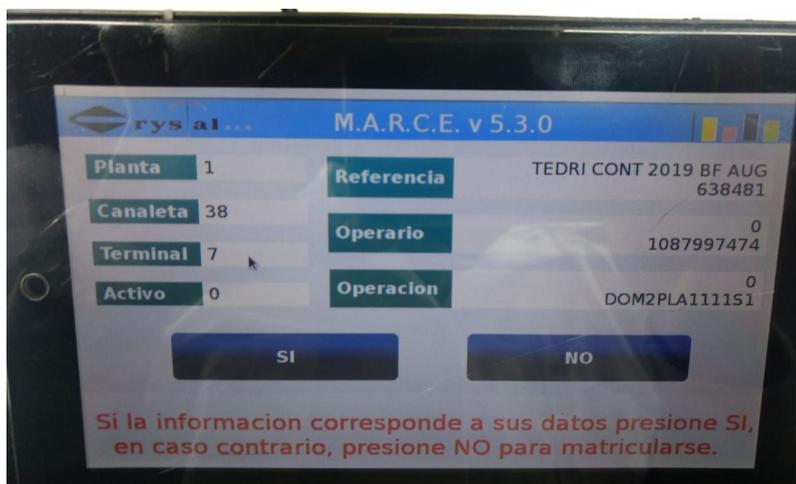


Figura 28. Pantalla final del inicio.

3.2.2 Pantalla matrícula física

Esta pestaña (ver **Figura 29**) corresponde a la matrícula física de la terminal, que indica su posición física en la planta de confección. Además, se añade la IP y la red en la cual está conectado el dispositivo.

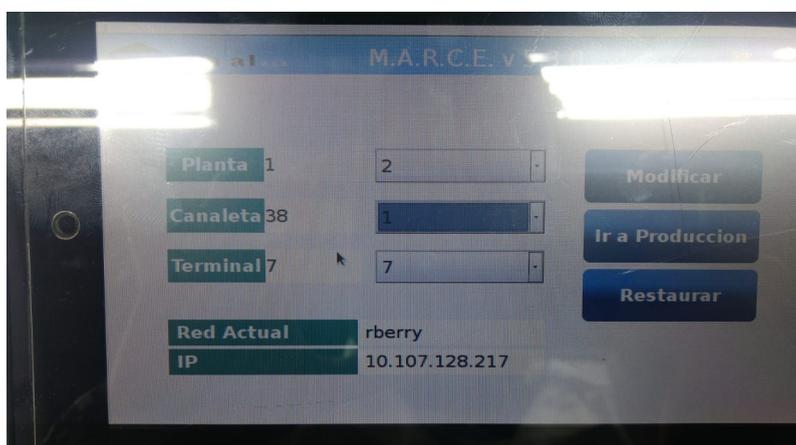


Figura 29. Pantalla final matrícula física.

3.2.3 Pantalla producción

Esta pantalla (ver **Figura 30**) muestra toda la información correspondiente a la operaria, a la operación y a la referencia que se están realizando en el momento de la matrícula y su desarrollo en producción. En esta pestaña podemos acceder a la mayoría de las opciones que contiene la terminal, como el cambio de operario, cambio de operación, ingreso de activo e inventario y reporte de paros. Aquí también se encuentra el botón auxiliar de conteo de la

producción que realiza la operaria cuando está matriculada. En esta pestaña se inicia la sesión, luego de terminar la matrícula, presionando el botón Iniciar sesión.



Figura 30. Pantalla final de producción.

3.2.4 Paros

La pantalla de paros (ver **Figura 31**) se compone de las opciones: paros, daños y causas que fueron determinados por un estudio previo liderado por el equipo de producción, en donde se parametrizan los daños más frecuentes en las máquinas que hacen que la producción pare.

Se debe escoger un tipo de paro y un daño para poder iniciar el cronómetro del paro.

Mientras que esté activada esta pestaña, la operaria no puede generar ningún conteo de prenda en la terminal.

Para finalizar el paro, se debe seleccionar previamente una causa y así parar el cronómetro y volver al estado de producción normal.



Figura 31. Pantalla final de paros.

3.2.5 Activo e inventario

Las pantallas de ingreso de activo e inventario (ver **Figura 32**) comparten la misma interfaz gráfica, la cual contiene un teclado numérico para ingresar el respectivo valor y dos botones que se encargan de confirmar en caso necesario el activo o volver a la ventana de producción sin guardar ningún cambio. Esta pestaña muestra también el activo o inventario ingresado previamente.



Figura 32. Pantalla final de activo e inventario.

3.2.6 Operario (selección del nombre de la operaria)

Esta pantalla (ver **Figura 33**) contiene la lista de los operarios que se encuentran en la plantilla correspondiente a la matrícula física de la terminal, teniendo como prioridad, la persona asignada a trabajar en la terminal matriculada.

Luego de escoger un operario, se presiona el botón confirmar operario para asegurar que la selección fue correcta.

En esta pestaña se encuentra el botón cerrar sesión, el cual detiene el cronómetro e indica que el operario matriculado en ese momento no va a seguir en producción en esa terminal.



Figura 33. Pantalla final de selección de operario.

3.2.7 Operación (selección de la operación a desempeñar)

Esta pantalla (ver **Figura 34**) contiene las operaciones que se van a realizar en la referencia correspondiente a la matrícula física de la terminal, teniendo como prioridad la operación que se va a hacer.

Esta pantalla contiene un botón de confirmación de la operación, para asegurar su selección correctamente, además de un botón para volver a producción sin guardar los cambios.

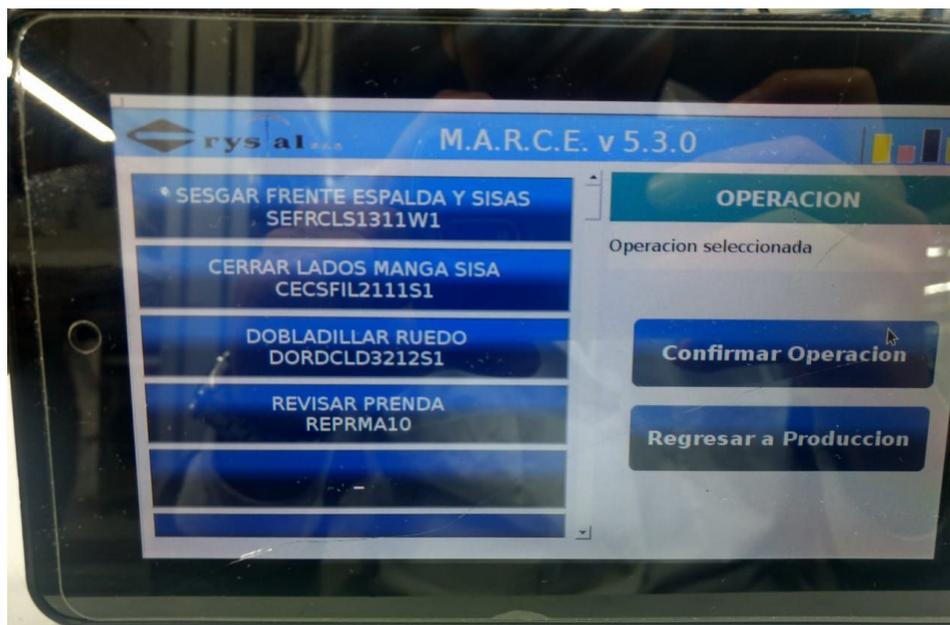


Figura 34. Pantalla final de selección de operación.

3.3 Uso de la aplicación por parte de los supervisores y analistas

Uno de los objetivos principales de la aplicación consiste en ayudar a las supervisoras y analistas en la toma de datos y de decisiones que permitan aumentar la producción en el transcurso del día, y no en largos periodos de tiempos como se estaba haciendo en la planta de confección con ayuda en un cronómetro.

Luego de unas semanas de uso de la aplicación por parte de las supervisoras y los analistas, se evidenció un ahorro significativo en el tiempo de toma de decisiones al momento de querer aumentar el potencial de un módulo específico, además permitió darse cuenta de errores y paros mucho más rápidamente lo que al final disminuyó el tiempo perdido en la jornada laboral.

CONCLUSIONES

En esta parte se presentan las conclusiones resultantes del desarrollo de la aplicación para cada una de las etapas del proyecto y su implementación.

Las conclusiones se presentan en 3 secciones que engloban el planteamiento, desarrollo y finalización.

1. Planteamiento del proyecto.

- Uno de los pasos más importantes para el desarrollo exitoso de un proyecto de ingeniería que busca la solución a un problema de un área específica, consiste en el estudio del área y de los procesos que se desarrollan en torno a ella. Por eso una de las etapas fundamentales del proyecto, fue el análisis de los requerimientos y el entendimiento de las dinámicas manejadas en el área de confección por parte de las personas y recursos.
- En el inicio del proyecto se deben plantear y especificar los objetivos generales y específicos del proyecto claramente, para evaluar el alcance y planear desde el inicio los recursos y el tiempo necesarios que se deben dedicar a cada uno de los puntos a desarrollar.
- Es de suma importancia que todas las partes involucradas en el proyecto participen en su planteamiento y desarrollo, ya que son estas las que van a obtener el producto final
- Al inicio se deben tener en cuenta las opiniones de cada uno de los grupos que van a usar el resultado final del proyecto en sus actividades diarias para que no se tenga que realizar cambios repentinos que atrasen o modifiquen el desarrollo del proyecto en una etapa tardía.

2. Ejecución del proyecto.

- Para un buen desarrollo del proyecto, es muy importante tener claros los objetivos generales y específicos, para realizar las actividades necesarias que puedan dar solución a los objetivos planteados en la etapa inicial del proyecto.
- El tiempo es muy importante a la hora de la ejecución del proyecto, ya que no basta solamente con tener claras las actividades a realizar, sino que también se debe establecer

el tiempo que se necesita en la realización de cada una de ellas. Por esto, es muy importante el planteamiento de un cronograma que contenga las tareas y el tiempo estipulado para ejecutar cada una de ellas, y así, asegurar un seguimiento temporal al avance del proyecto, así como prevenir futuros retrasos en su finalización.

- Es muy importante que el desarrollo del proyecto se haga de forma modular, con el fin de tener tareas pequeñas que sean fáciles de resolver, y se pueda hacer un correcto seguimiento al desarrollo y finalización de las mismas. Además se puede hacer una mejor implementación de la tarea para que cumpla una función específica y en futuros cambios, sea mucho más fácil hacer mantenimiento al proyecto final.

3. Conclusiones técnicas del sistema.

- La arquitectura cliente-servidor utilizada en este proyecto, fue escogida debido a que muchas de las aplicaciones de la empresa fueron desarrolladas con base en esta arquitectura y, por lo tanto, se disponía de un conocimiento previo de cómo desarrollar correctamente un proyecto con base en esta arquitectura. Adicionalmente esto permitió la centralización adecuada de los datos y su fácil réplica en las plantas de producción.
- El protocolo de comunicación *UDP*, aunque no asegura la entrega de los mensajes, permitió no saturar el tráfico que se genera en la red cuando hay muchos dispositivos conectados al mismo tiempo y que intentan comunicarse con el servidor. La aplicación mostró un gran desempeño en asegurar que los mensajes llegaran correctamente.
- El dispositivo escogido para el desarrollo de la aplicación, *la Raspberry pi*, ofreció bastantes facilidades a la hora del diseño e implementación de la aplicación. Al tener la facilidad de programar la aplicación en un lenguaje poco pesado, y tener la posibilidad de conectar varios periféricos, se logró la conexión de un pulsador externo seguro y una aplicación ágil y amigable con el usuario.
- El trabajo con las operarias, supervisores y analistas, fue demasiado importante al momento del desarrollo y uso de la aplicación.
- El entender los datos y cuál era su uso en producción, ayudó en un desarrollo y acondicionamiento mejor de la aplicación para su uso adecuado en las plantas de confección.

REFERENCIAS

1. Quienes somos, Crystal sas. crystal. [Internet] [Consultado 2019 mayo 21] disponible en: http://www.crystal.com.co/es/quienes_somos
2. Python .Wikipedia.[Internet] [Consultado 2019 mayo 21] disponible en: <https://es.wikipedia.org/wiki/Python>
3. Python .net. [Internet] [Consultado 2019 mayo 21] disponible en: <https://.net/python/>
4. Why python was created in the first place. Python org.[Internet] [Consultado 2019 mayo 21] disponible

en:<https://docs.python.org/3/faq/general.html#why-was-python-created-in-the-first-place>

5. Python. Lenguajes de programación. [Internet] [Consultado 2019 mayo 21] disponible en:
<https://lenguajesdeprogramacion.net/python/>
6. Ventajas que nos ofrece Python respecto a otros lenguajes. quora.[Internet] [Consultado 2019 mayo 21] disponible
en:<https://es.quora.com/Qu%C3%A9-ventajas-nos-ofrece-Python-respecto-a-otros-lenguajes-de-programaci%C3%B3n>
7. Ventajas y desventajas de Python. entrenamiento Python básico.[Internet] [Consultado 2019 mayo 21] disponible
en:<https://entrenamiento-python-basico.readthedocs.io/es/latest/leccion1/ventajas-desventajas.html>
8. Raspberry pi. Wikipedia.[Internet] [Consultado 2019 mayo 21] disponible en:
https://es.wikipedia.org/wiki/Raspberry_Pi
9. Review Raspberry pi 3. Wikipedia. [Internet] [Consultado 2019 mayo 21] disponible
en:<https://hardzone.es/2018/07/02/review-raspberry-pi-3-modelo-b/>
10. Raspberry pi primeros pasos. vuxmi.[Internet] [Consultado 2019 mayo 21] disponible en:
<http://vuxmi.com/raspberry-pi-primeros-pasos/>
11. Protocolo de datagrama de usuario. Wikipedia.[Internet] [Consultado 2019 mayo 21] disponible en:https://es.wikipedia.org/wiki/Protocolo_de_datagramas_de_usuario
12. Protocolo de control de transmisión. Wikipedia. [Internet] [Consultado 2019 mayo 21] disponible en:https://es.wikipedia.org/wiki/Protocolo_de_control_de_transmisi%C3%B3n
13. UDP Communication. Wikipedia.[Internet] [Consultado 2019 mayo 21] disponible en:
<https://wiki.python.org/moin/UdpCommunication>
14. Python socket. Python org.[Internet] [Consultado 2019 mayo 21] disponible en:
<https://docs.python.org/3/library/socket.html>
15. Las carpetas compartidas.platea.pntic.mec.es.[Internet] [Consultado 2019 mayo 21] disponible
en:http://platea.pntic.mec.es/vgonzale/trabcolab_0910/archivos/_110/Tema_1.2.htm
16. Carpetas compartidas. Portal uah.[Internet] [Consultado 2019 mayo 21] disponible
en:https://portal.uah.es/portal/page/portal/servicios_informaticos/catalogo/hospedaje/Trabajo_en_grupo/Carpetas_compartidas
17. What is pyqtgraph. [Internet] [Consultado 2019 mayo 21] disponible
en:<http://www.pyqtgraph.org/documentation/introduction.html#what-is-pyqtgraph>
18. qt for Python. qt.[Internet] [Consultado 2019 mayo 21] disponible en:
<https://www.qt.io/qt-for-python>

19. Definición de GUI. sistemas.[Internet] [Consultado 2019 mayo 21] disponible en:
<https://sistemas.com/gui.php>
20. Modelo vista controlador. Wikipedia.[Internet] [Consultado 2019 mayo 21] disponible en:
<https://es.wikipedia.org/wiki/Modelo%28%93vista%28%93controlador>
21. Cuantil. Wikipedia.[Internet] [Consultado 2019 mayo 21] disponible en:
<https://es.wikipedia.org/wiki/Cuantil>