



Controlador CAN de Servomotores

Autor:

Alejandro Virgillo

Director:

Gabriel Gavinowich (FIUBA)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 21 de octubre de 2021 y el 9 de diciembre de 2021.*

Índice

1. Descripción técnica-conceptual del proyecto a realizar	5
2. Identificación y análisis de los interesados	6
3. Propósito del proyecto	7
4. Alcance del proyecto	7
5. Supuestos del proyecto.	7
6. Requerimientos	7
7. Historias de usuarios (<i>Product backlog</i>).	9
8. Entregables principales del proyecto	11
9. Desglose del trabajo en tareas	11
10. Diagrama de Activity On Node.	12
11. Diagrama de Gantt	13
12. Presupuesto detallado del proyecto	15
13. Gestión de riesgos	15
14. Gestión de la calidad	16
15. Procesos de cierre	17

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	21 de octubre de 2021
1	Se completa hasta el punto 5 inclusive	30 de octubre de 2021
2	Se completa hasta el punto 9 inclusive	7 de noviembre de 2021

Acta de constitución del proyecto

Buenos Aires, 21 de octubre de 2021

Por medio de la presente se acuerda con el Ing. Alejandro Virgillo que su Trabajo Final de la Carrera de Especialización en Sistemas Embebidos se titulará “Controlador CAN de Servomotores”, consistirá esencialmente en la implementación de una interfaz que permita configurar y relevar información de servomotores conectados a través de una red CAN, y tendrá un presupuesto preliminar estimado de 600 hs de trabajo y U\$1000, con fecha de inicio 21 de octubre de 2021 y fecha de presentación pública 9 de octubre de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg
Director posgrado FIUBA

Alejandro Virgillo
A3 Engineering

Gabriel Gavinowich
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

El proyecto busca obtener un dispositivo que permita programar y supervisar servomotores conectados dentro de una red CAN (*Controller Area Network*), así como relevar información de estos. El resultado debe ser de carácter industrial, por lo que se prioriza su robustez para poder operar en planta y debe contar con una interfaz de usuario.

La organización A3 Engineering desarrolló un sistema embebido, llamado SN-17, capaz de controlar la posición, velocidad, aceleración y torque a motores del tipo paso a paso, también conocidos como *steppers*. Actualmente, varios de estos motores, junto con las placas controladoras mencionadas, se encuentran en funcionamiento en la planta de la empresa Cambre, realizando diversos tipos de actuaciones mecánicas en líneas de manufactura.

El sistema SN-17 cuenta con un problema, que es la dificultad para alterar, de forma cómoda, el funcionamiento de los motores. El firmware debe ser modificado cada vez que se quiera realizar un cambio de este estilo, lo cual limita la operabilidad.

Dentro de este contexto es que se propone el actual proyecto. Se buscará lograr una interfaz que permita a un usuario con poco entrenamiento conectarse con los motores y modificar los parámetros de funcionamiento. Para ello, se establecerá comunicación a través de una red CAN, aprovechando que el sistema SN-17 cuenta con un puerto para este protocolo. Es necesario, entonces, diseñar un sistema de mensajería que haga que los dispositivos puedan enviar información dentro de la red, y una interfaz gráfica que permita a usuarios realizar cambios.

También se propone que el dispositivo actúe como supervisor, una vez que los motores estén en funcionamiento. Como en la mayoría de procesos industriales suelen emplearse controladores PLC (*Programmable Logic Controller*), es necesario que la solución buscada pueda interactuar con estos.

En la **Figura 1** se muestra, a modo de ejemplo, un diagrama en bloques del sistema. El proyecto abarca el diseño y fabricación de la parte que aparece denominada como controlador y su interacción con los demás bloques. También notar que en el bus CAN pueden haber conectados más de 1 servomotor. Cada uno de estos tendrá una de las placas controladoras descriptas previamente.

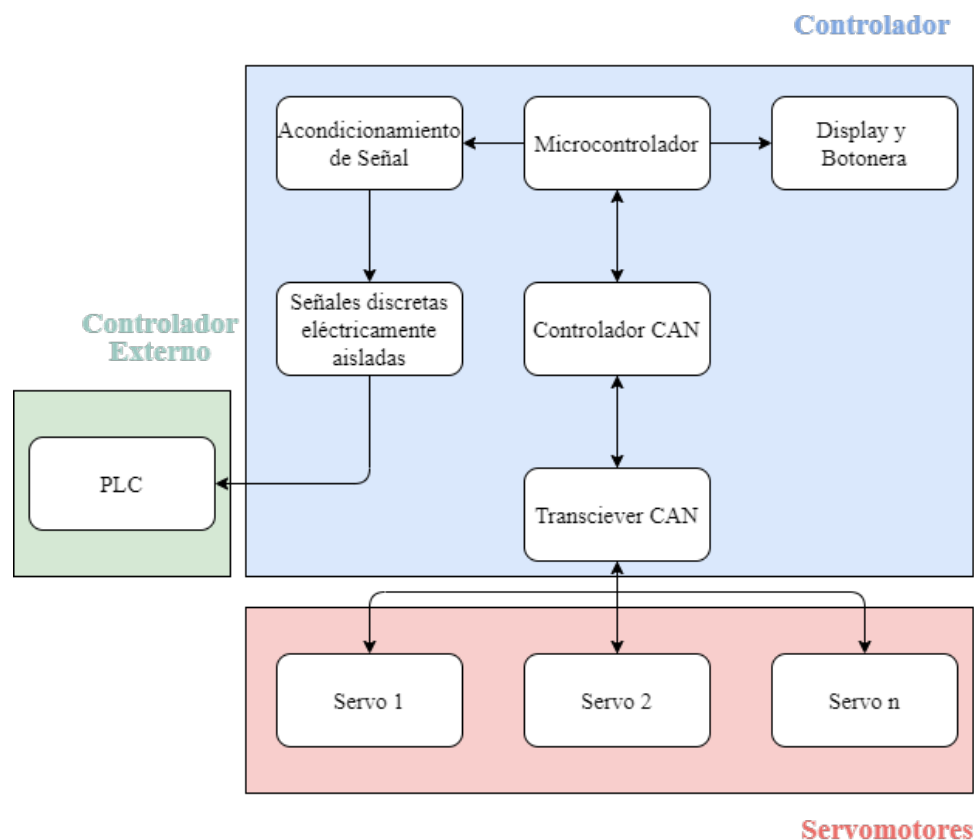


Figura 1. Diagrama en bloques del sistema

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Auspiciante	Alejandro Virgillo	A3 Engineering	Líder de proyecto
Cliente	Alejandro Virgillo	A3 Engineering	Líder de proyecto
Impulsor	Sector de automatización	Cambre	-
Responsable	Alejandro Virgillo	FIUBA	Alumno
Colaboradores	Andres Battisti	Cambre	Jefe de Automatización
Orientador	Gabriel Gavinowich	FIUBA	Director Trabajo final
Usuario final	Sector de automatización	Cambre	Técnicos
Usuario final	Sector de armado	Cambre	Operarios

Por ejemplo:

- Orientador: Gabriel Gavinowich, es especialista en sistemas embebidos y trabaja en protocolo CAN, será de gran ayuda en materias de este aspecto.
- Colaborador: Andrés Battisti, es hábil en la coordinación de proyectos. Puede colaborar con la implementación en planta.
- Usuario final: Sería de ayuda consultar a los operarios del sector de automatización para determinar temas de uso y de calidad que puedan considerar deseable. Pueden dar pautas para requerimientos que pueden ser de utilidad.

3. Propósito del proyecto

El propósito de este proyecto es desarrollar un sistema embebido que actúe de interfaz para configurar y supervisar servomotores conectados a una red CAN.

4. Alcance del proyecto

El proyecto incluye:

- Una interfaz de usuario que permite configurar y supervisar los servomotores conectados.
- La estructura de mensajes que ha de transmitirse a través del bus CAN.
- La inclusión de entradas y salidas eléctricamente aisladas para comunicación con un PLC.
- La configuración de la red CAN.
- El desarrollo y fabricación de una plaqueta que abarque al sistema.

El proyecto no incluye:

- El acceso a los datos de funcionamiento de los servomotores de forma remota o el almacenamiento de estos en una memoria.
- El desarrollo de las placas controladoras de los servomotores.
- La implementación final en planta.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- El dinero disponible será suficiente para la adquisición de los materiales requeridos.
- Habrá stock de los componentes del sistema y no habrá problemas de importación.
- Las demoras para obtener los componentes no serán excesivas.
- La relación con la empresa Cambre se mantendrá durante el transcurso del proyecto.

6. Requerimientos

1. Requerimientos de la red CAN:

- 1.1. El sistema debe comunicarse empleando el protocolo CAN. [SCI-CAN-REQ001]
- 1.2. El sistema debe poder comunicarse con todos los motores conectados a la red CAN que posean la placa controladora SN-17. [SCI-CAN-REQ002]

- 1.3. El sistema debe poder comunicarse con hasta 5 motores conectados a la red CAN. [SCI-CAN-REQ003]
- 1.4. El sistema debe enviar y recibir información a través del puerto CAN a una tasa de al menos 300 kbps. [SCI-CAN-REQ004]

2. Requerimientos de comunicación:

- 2.1. El sistema debe poder enviar programas a los servomotores de hasta 15 instrucciones de largo. [SCI-CAN-REQ005]
- 2.2. El sistema debe tener la capacidad de enviar cada una de las instrucciones de programa disponibles en los servomotores. Estas son: Seteo de tipo de control, verificación de señal de control, pausa de la señal de control, apagado de control, comunicación, reseteo de programa. [SCI-CAN-REQ006]
- 2.3. El sistema debe enviar, para cada instrucción, todos los atributos de configuración necesarios. Estos son: Selección de variable de control, seteo de variable de control, tiempo de cumplimiento de instrucción, tiempo de timeout, error permisible de la variable de control, torque máximo de motor, tipo de comunicación y mensaje. [SCI-CAN-REQ007]
- 2.4. El sistema debe poder modificar los parámetros de los lazos de control PID de los servomotores para los distintos tipos de control. [SCI-CAN-REQ008]
- 2.5. El sistema debe poder configurar el seteo de funcionalidades especiales de los motores, estas son: selección de tipo de cerado, proceso de cerado, calibración del encoder, calibración de posiciones, forma de operatividad de las salidas discretas de los motores, verificación de detención de eje. [SCI-CAN-REQ009]
- 2.6. El sistema debe permitir operar los motores en forma manual, simulando las entradas y forzando las salidas discretas que cada uno de los motores conectados. [SCI-CAN-REQ010]
- 2.7. El sistema deberá poder activar o desactivar a cada uno de los motores conectados. [SCI-CAN-REQ011]
- 2.8. El sistema deberá recibir de cada motor conectado el número de programa e instrucción en que se encuentra. [SCI-CAN-REQ012]
- 2.9. El sistema deberá recibir de cada motor conectado un reporte de error en caso de falla. [SCI-CAN-REQ013]

3. Requerimientos de comunicación con PLC:

- 3.1. El sistema debe poder comunicarse con un controlador externo (PLC) a través de señales discretas. [SCI-CAN-REQ014]
- 3.2. Las señales al controlador externo deben estar eléctricamente aisladas, por lo menos a 1 kV, usando optoacopladores. [SCI-CAN-REQ015]
- 3.3. Las señales al controlador externo deben ser del tipo NPN. [SCI-CAN-REQ016]
- 3.4. Las señales al controlador externo deben poder ser a distinta tensión que la empleada por el microcontrolador, hasta 24V. [SCI-CAN-REQ017]
- 3.5. El sistema deberá enviar señales discretas en caso de que alguno de los motores conectados reporte un error. [SCI-CAN-REQ018]

4. Requerimientos de interfaz de usuario:

- 4.1. El sistema debe tener una pantalla LCD y una botonera. [SCI-CAN-REQ019]

- 4.2. La pantalla LCD debe permitir acceder a las variables de los motores a través de un menú y debe recorrerse usando la botonera. [SCI-CAN-REQ020]
- 4.3. El sistema debe tener un switch que permita cambiar su forma de funcionamiento, entre configurador de motores y relevador de información. [SCI-CAN-REQ021]

5. Requerimientos de diseño:

- 5.1. El sistema debe emplear el microcontrolador ATSAMC21. [SCI-CAN-REQ022]
- 5.2. El sistema debe alimentarse con 24V de tensión. [SCI-CAN-REQ023]
- 5.3. El sistema debe estar encapsulado dentro de una caja plástica. [SCI-CAN-REQ024]

7. Historias de usuarios (*Product backlog*)

En esta sección se enuncian las historias de usuario, cada una de ellas llevará un puntaje según 3 aspectos:

- Dificultad: Cantidad de trabajo a realizar.
- Complejidad: Complejidad de trabajo a realizar.
- Riesgo: Incertidumbre del trabajo a realizar.

Se utilizará una escala siguiendo la serie de Fibonacci, donde un número mayor implica mayor costo. Si la suma de los 3 componentes no da un número de la serie, se elegirá el próximo más cercano.

1. Como operario quiero detectar la presencia de errores para reportarlo a mi supervisor.
 - D: 8.
 - C: 5.
 - R: 3.
 - Total: 21.
2. Como operario quiero una interfaz de usuario simple para evitar cometer errores.
 - D: 3.
 - C: 1.
 - R: 1.
 - Total: 5.
3. Como trabajador de mantenimiento quiero conocer los problemas de los actuadores para repararlos rápidamente.
 - D: 5.
 - C: 5.
 - R: 3.
 - Total: 13.

4. Como desarrollador quiero controlar los actuadores para facilitar la programación.
 - D: 3.
 - C: 5.
 - R: 3.
 - Total: 13.
5. Como desarrollador quiero saber el estado de funcionamiento para facilitar la programación.
 - D: 8.
 - C: 5.
 - R: 3.
 - Total: 21.
6. Como programador de PLC quiero recibir señales del estado de actuadores para facilitar la programación.
 - D: 3.
 - C: 3.
 - R: 1.
 - Total: 8.
7. Como gerente de planta, quiero minimizar la cantidad de sensores para disminuir los costos.
 - D: 3.
 - C: 3.
 - R: 3.
 - Total: 13.
8. Como gerente de planta, quiero que los componentes de las líneas de producción sean robustos para minimizar los tiempos de parada.
 - D: 3.
 - C: 5.
 - R: 5.
 - Total: 13.
9. Como gerente de planta, quiero que los actuadores de las líneas reporten errores para minimizar mantenimiento.
 - D: 5.
 - C: 5.
 - R: 3.
 - Total: 13.

8. Entregables principales del proyecto

Los entregables del proyecto son:

- Prototipo del sistema
- Manual de uso
- Diagrama de circuitos esquemáticos
- Diagrama de PCB
- Archivos de fabricación de PCB
- Diagrama de instalación
- Planos de caja
- Informe final

9. Desglose del trabajo en tareas

A continuación se enumeran las tareas del proyecto y se detalla su carga horaria:

1. Planificación y gestión del proyecto (80 hs):

- 1.1. Realizar el plan de trabajo (20 hs).
- 1.2. Determinación de componentes (10 hs)
- 1.3. Realizar informes de avance (10 hs)
- 1.4. Confección de memoria de trabajo (30 hs)
- 1.5. Presentación y defensa de trabajo (10 hs)

2. Tareas de investigación (60 hs):

- 2.1. Búsqueda de soluciones similares (20 hs)
- 2.2. Estudiar protocolo CAN (20 hs)
- 2.3. Examinar prestaciones del microcontrolador (10 hs)
- 2.4. Recopilar hojas de datos de componentes y búsqueda de librerías (10 hs)

3. Desarrollo de software (185 hs):

- 3.1. Integración de drivers y librerías (15 hs)
- 3.2. Elaboración de estructura de mensajes (30 hs)
- 3.3. Programación de red CAN (20 hs)
- 3.4. Implementación de estructura de mensajes (20 hs)
- 3.5. Programación de menús (20 hs)
- 3.6. Desarrollo de aplicación (40 hs)
- 3.7. Integración de software de servomotores (40 hs)

4. Desarrollo de Hardware (95 hs):

- 4.1. Diseño de circuitos eléctricos (30 hs)
- 4.2. Diseño de PCB (30 hs)
- 4.3. Diseño de red CAN (20 hs)
- 4.4. Elaboración de archivos de fabricación (15 hs)

5. Fabricación (65 hs):

- 5.1. Compra de componentes (20 hs)
- 5.2. Fabricación de PCB (5 hs)
- 5.3. Ensamble de PCB (20 hs)
- 5.4. Diseño y fabricación de caja (20 hs)

6. Ensayos e implementación(145 hs):

- 6.1. Pruebas eléctricas (15 hs)
- 6.2. Ensayo de comunicación (40 hs)
- 6.3. Ensayo de interfaz de usuario (20 hs)
- 6.4. Prueba de aplicación (30 hs)
- 6.5. Optimización y búsqueda de errores (20 hs)
- 6.6. Pruebas de implementación en planta (20 hs)

Cantidad total de horas: (630 hs)

10. Diagrama de Activity On Node

Armar el AoN a partir del WBS definido en la etapa anterior.

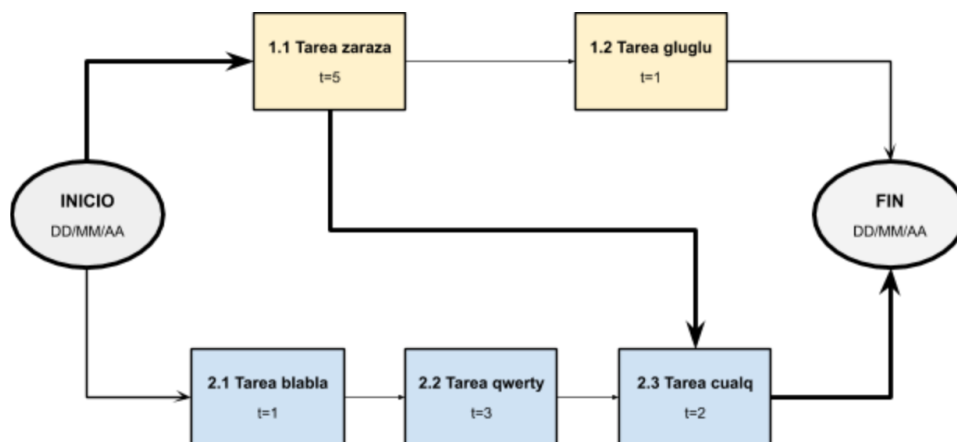


Figura 2. Diagrama en *Activity on Node*

Indicar claramente en qué unidades están expresados los tiempos. De ser necesario indicar los caminos semicríticos y analizar sus tiempos mediante un cuadro. Es recomendable usar colores y un cuadro indicativo describiendo qué representa cada color, como se muestra en el siguiente ejemplo:

11. Diagrama de Gantt

Existen muchos programas y recursos *online* para hacer diagramas de gantt, entre los cuales destacamos:

- Planner
- GanttProject
- Trello + *plugins*. En el siguiente link hay un tutorial oficial:
<https://blog.trello.com/es/diagrama-de-gantt-de-un-proyecto>
- Creately, herramienta online colaborativa.
<https://creately.com/diagram/example/ieb3p3ml/LaTeX>
- Se puede hacer en latex con el paquete *pgfgantt*
<http://ctan.dcc.uchile.cl/graphics/pgf/contrib/pgfgantt/pgfgantt.pdf>

Pegar acá una captura de pantalla del diagrama de Gantt, cuidando que la letra sea suficientemente grande como para ser legible. Si el diagrama queda demasiado ancho, se puede pegar primero la “tabla” del Gantt y luego pegar la parte del diagrama de barras del diagrama de Gantt.

Configurar el software para que en la parte de la tabla muestre los códigos del EDT (WBS).
Configurar el software para que al lado de cada barra muestre el nombre de cada tarea.
Revisar que la fecha de finalización coincida con lo indicado en el Acta Constitutiva.

En la figura 3, se muestra un ejemplo de diagrama de gantt realizado con el paquete de *pgfgantt*. En la plantilla pueden ver el código que lo genera y usarlo de base para construir el propio.

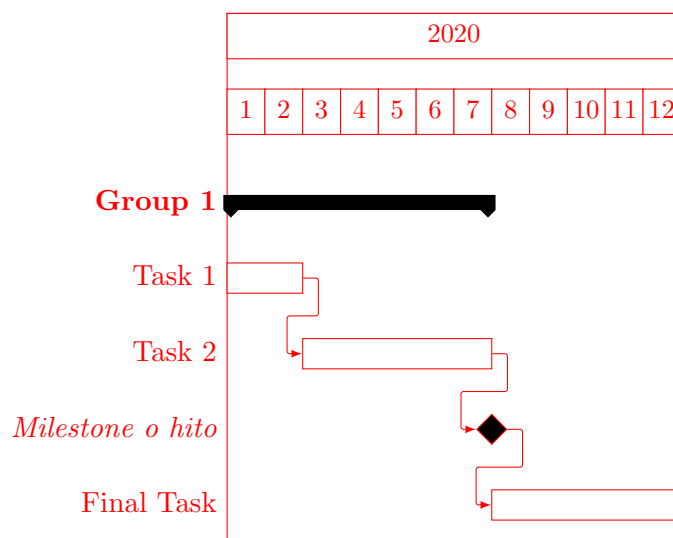


Figura 3. Diagrama de gantt de ejemplo



Figura 4. Ejemplo de diagrama de Gantt rotado

12. Presupuesto detallado del proyecto

Si el proyecto es complejo entonces separarlo en partes:

- Un total global, indicando el subtotal acumulado por cada una de las áreas.
- El desglose detallado del subtotal de cada una de las áreas.

IMPORTANTE: No olvidarse de considerar los **COSTOS INDIRECTOS**.

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
SUBTOTAL			
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
SUBTOTAL			
TOTAL			

13. Gestión de riesgos

a) Identificación de los riesgos (al menos cinco) y estimación de sus consecuencias:

Riesgo 1: detallar el riesgo (riesgo es algo que si ocurre altera los planes previstos de forma negativa)

- Severidad (S): mientras más severo, más alto es el número (usar números del 1 al 10). Justificar el motivo por el cual se asigna determinado número de severidad (S).
- Probabilidad de ocurrencia (O): mientras más probable, más alto es el número (usar del 1 al 10). Justificar el motivo por el cual se asigna determinado número de (O).

Riesgo 2:

- Severidad (S):
- Ocurrencia (O):

Riesgo 3:

- Severidad (S):

■ Ocurrecia (O):

b) Tabla de gestión de riesgos: (El RPN se calcula como $RPN=S \times O$)

Riesgo	S	O	RPN	S*	O*	RPN*

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a...

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

Riesgo 1: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación). Nueva asignación de S y O, con su respectiva justificación: - Severidad (S): mientras más severo, más alto es el número (usar números del 1 al 10). Justificar el motivo por el cual se asigna determinado número de severidad (S). - Probabilidad de ocurrencia (O): mientras más probable, más alto es el número (usar del 1 al 10). Justificar el motivo por el cual se asigna determinado número de (O).

Riesgo 2: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación).

Riesgo 3: plan de mitigación (si por el RPN fuera necesario elaborar un plan de mitigación).

14. Gestión de la calidad

Para cada uno de los requerimientos del proyecto indique:

- Req #1: copiar acá el requerimiento.
 - Verificación para confirmar si se cumplió con lo requerido antes de mostrar el sistema al cliente. Detallar
 - Validación con el cliente para confirmar que está de acuerdo en que se cumplió con lo requerido. Detallar

Tener en cuenta que en este contexto se pueden mencionar simulaciones, cálculos, revisión de hojas de datos, consulta con expertos, mediciones, etc. Las acciones de verificación suelen considerar al entregable como “caja blanca”, es decir se conoce en profundidad su funcionamiento interno. En cambio, las acciones de validación suelen considerar al entregable como “caja negra”, es decir, que no se conocen los detalles de su funcionamiento interno.

15. Procesos de cierre

Establecer las pautas de trabajo para realizar una reunión final de evaluación del proyecto, tal que contemple las siguientes actividades:

- Pautas de trabajo que se seguirán para analizar si se respetó el Plan de Proyecto original:
- Indicar quién se ocupará de hacer esto y cuál será el procedimiento a aplicar.
- Identificación de las técnicas y procedimientos útiles e inútiles que se emplearon, y los problemas que surgieron y cómo se solucionaron: - Indicar quién se ocupará de hacer esto y cuál será el procedimiento para dejar registro.
- Indicar quién organizará el acto de agradecimiento a todos los interesados, y en especial al equipo de trabajo y colaboradores: - Indicar esto y quién financiará los gastos correspondientes.