



Sistema de monitoreo y control de ambientes a distancia

Autor:

Ing. César Javier Fanelli

Director:

Ing. Fernando Lichtschein (FIUBA)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 21 de octubre de 2022 y el 8 de diciembre de 2022.*

Índice

1. Descripción técnica-conceptual del proyecto a realizar	5
2. Identificación y análisis de los interesados	6
3. Propósito del proyecto	6
4. Alcance del proyecto	7
5. Supuestos del proyecto.	7
6. Requerimientos	8
7. Historias de usuarios (<i>Product backlog</i>).	9
8. Entregables principales del proyecto	9
9. Desglose del trabajo en tareas	9
10. Diagrama de Activity On Node.	11
11. Diagrama de Gantt	12
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
1.0	Creación del documento	20/10/2022
1.1	Se completa hasta la sección 5 inclusive	3/11/2022
1.2	Se corrige hasta la sección 5 y se completa hasta la sección 9 inclusive	10/11/2022

Acta de constitución del proyecto

Buenos Aires, 21 de octubre de 2022

Por medio de la presente se acuerda con el Ing. Ing. César Javier Fanelli que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará “Sistema de monitoreo y control de ambientes a distancia”, consistirá esencialmente en el diseño e implementación de un prototipo un sistema de control de ambientes por internet, y tendrá un presupuesto preliminar estimado de 605 hs de trabajo y USD850, con fecha de inicio 21 de octubre de 2022 y fecha de presentación pública a definir.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg
Director posgrado FIUBA

Mg. Ing. Damián Corbalán
Movistar

Ing. Fernando Lichtschein
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

Es sabido que la domótica, el monitoreo y el control de parámetros dentro de un ambiente está creciendo a grandes pasos. Tanto es así que empresas grandes como Google, Apple o Amazon desarrollaron sus propios sistemas de hogares inteligentes. Inclusive, están desarrollando en conjunto con otras empresas un sistema que sea compatible entre esas marcas, llamado Matter, de código abierto.

Así se puede ver el interés y hasta a veces la necesidad de conocer el estado y poder manejar parámetros a distancia de habitaciones, oficinas, salas de estar, y otros recintos. De esta manera se facilita la vida cotidiana y se pueden generar situaciones de confort con poco esfuerzo, dedicando el tiempo y la energía en otros aspectos de la vida cotidiana. Tan sólo con un click o pulsar una pantalla, se conocen o modifican los estados del sector que se desee.

Este proyecto, de índole personal o académica, surge de la idea principal de la tesis de grado de la carrera de la cual egresé en 2014 de ingeniero electrónico. El modelo anterior de sistema, era un prototipo básico de control inteligente de hogares, que si bien poseía comunicación inalámbrica y control de varios actuadores, carecía de varios aspectos, como ser, poder hacer el control mediante una PC o smartphone y la conexión a internet, que le aporta funcionalidad y versatilidad.

En la figura 1 se muestra un diagrama en bloques del sistema a desarrollar en el proyecto. Se observa que consta de 2 partes físicas diferenciables: el nodo, o actuador remoto, y el servidor backend, implementado en una Raspberry Pi. El nodo a su vez estará compuesto por el módulo de control (comandado por una placa de desarrollo con ESP32), un display para indicar los valores medidos y seteados, un potenciómetro digital para variar la temperatura, una salida de relé para el calefactor y una salida PWM para controlar una luminaria LED de 12 VCC. El usuario podrá acceder mediante una página web, con usuario y contraseña, a visualizar y cambiar los parámetros de la habitación donde esté instalado el sistema. En este caso, el sistema se implementará y probará con un solo nodo.

La diferencia con los sistemas ya existentes, es que sea fácilmente adaptable a entornos fuera de un hogar y una oficina, tales como una sala de servidores, un área limpia de un laboratorio, u otros lugares donde se requiera sensar distintos parámetros y poder además tener un historial de ellos.

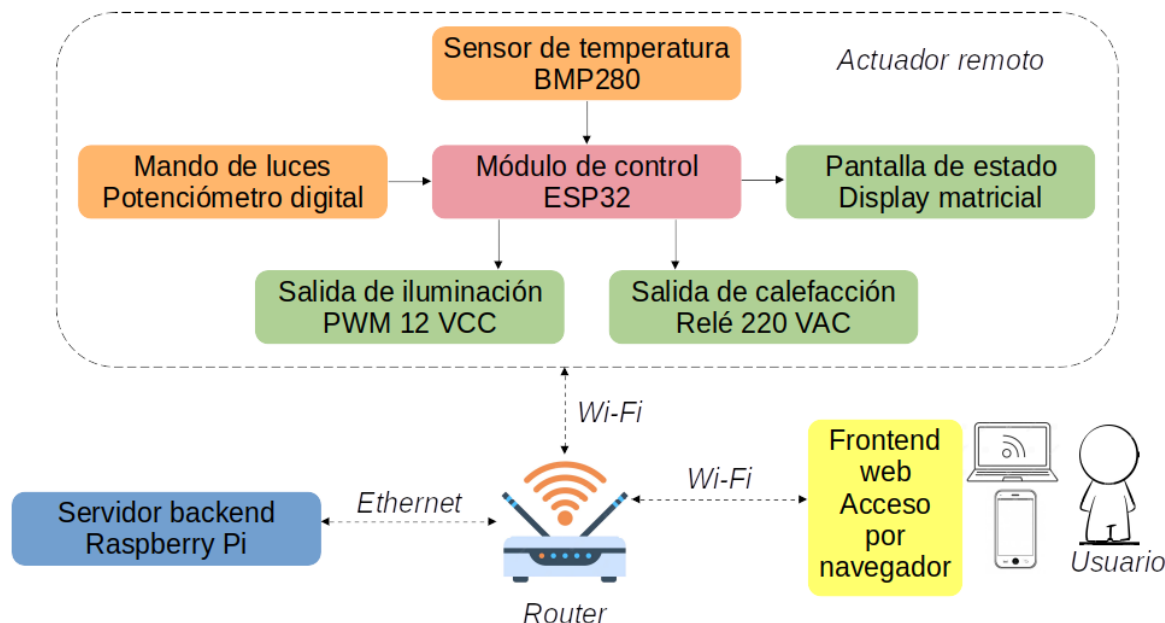


Figura 1. Diagrama en bloques del sistema.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Mg. Ing. Damián Corbalán	Movistar	Especialista de redes IP
Responsable	Ing. César Javier Fanelli	FIUBA	Alumno
Orientador	Ing. Fernando Lichtscheim	FIUBA	Director Trabajo final
Usuario final	Usuarios que habiten o utilicen los recintos	-	-

- Cliente: es una persona exigente con muchos conocimientos en el tema.
- Responsable: Ing. César Javier Fanelli, único responsable a cargo del desarrollo del proyecto. Se ocupará de la planificación y ejecución de las tareas.
- Orientador: va a poder colaborar mucho con el desarrollo del proyecto y la resolución de los problemas que surjan en el camino.

3. Propósito del proyecto

El propósito de este proyecto es crear un prototipo de sistema de ambientes inteligentes, que sirva como puntapié inicial de un sistema más grande y más completo. Al ser personal la intención es diseñar un sistema que sea potencialmente escalable, al que se le puedan ir agregando más módulos con distintas funcionalidades que sean de utilidad en la vida cotidiana. Dichas funciones futuras podrían ser controlar el acceso (que se lea desde un celular), integrar cámaras

de seguridad o sensor y modificar de otros parámetros. El aprendizaje y el desafío son las motivaciones para hacer un proyecto de esta índole. Y si resultara siendo viable técnica y económicamente, en un futuro se analizará la posibilidad de un desarrollo de negocio.

4. Alcance del proyecto

El proyecto en cuestión incluye:

- Desarrollo y fabricación de un prototipo de placa que actúe una salida ON-OFF para controlar el calefactor y otra salida para la iluminación.
- Desarrollo e implementación de las conexiones entre la placa del punto anterior y la placa de desarrollo con el ESP32 (formando en conjunto el nodo).
- Desarrollo del software del nodo.
- Desarrollo del software del servidor.

El proyecto no incluye:

- Otros nodos cuyas funciones no sean las del ejemplo descrito en el diagrama en bloques.
- Software en Android o iOS para acceder a la plataforma.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- Se podrán conseguir todos los materiales necesarios para el armado del prototipo.
- Se contará con los recursos económicos para desarrollarlo con normalidad.
- No surgirán imprevistos personales que retrasen significativamente o pospongan el desarrollo del proyecto o la cursada del posgrado.
- Se adquirirán los conocimientos necesarios para la implementación del software y el sistema en su integridad.
- Se tendrá apoyo del director en temas netamente técnicos para el desarrollo.
- Se tendrá feedback constante del cliente para definir alcance, requerimientos y aprobar la funcionalidad del prototipo.
- El cliente cuenta con conexión a Internet y un hardware mínimo para conectar el sistema a su red local.

6. Requerimientos

Los requerimientos revisados en conjunto con el cliente y agrupados por afinidad son:

1. Requerimientos funcionales del sistema

- 1.1. El sistema deberá poder funcionar de forma automática o manual
- 1.2. El modo automático consistirá en poder programar horarios y niveles de las salidas, seteadas a través de la interfaz web
- 1.3. El modo manual consistirá en accionar las salidas desde la interfaz web o desde los comandos en el nodo
- 1.4. Al modificarse de forma manual cualquier parámetro, el nodo informará al servidor el nuevo valor del parámetro seteado
- 1.5. Al activar el modo manual, se interrumpirá el modo automático
- 1.6. Se guardarán los valores sensados y seteados en una base de datos dentro del servidor
- 1.7. Los usuarios podrán revisar los datos de los valores actuales e históricos

2. Requerimientos asociados al nodo

- 2.1. El nodo estará implementado en una placa con un ESP32.
- 2.2. Contará con un potenciómetro digital para elegir la temperatura e iluminación.
- 2.3. Tendrá con un display que mostrará la temperatura sensada y el estado de las salidas
- 2.4. La frecuencia de sensado de la temperatura será de 1 minuto
- 2.5. Los niveles de tensión de las entradas y salidas lógicas serán adaptados a los valores lógicos correspondientes de los actuadores
- 2.6. Se alimentará con una fuente de 5 VCC y un cable microUSB

3. Requerimientos asociados al servidor

- 3.1. El servidor estará montado en una Raspberry Pi con sistema operativo Linux
- 3.2. Alojará el código de la página web desde la que se ingresará al sistema
- 3.3. Alojará el código del backend del servidor y la base de datos
- 3.4. Se alimentará con una fuente de 5 VCC

4. Requerimientos de la interfaz

- 4.1. Será intuitiva y sencilla de operar
- 4.2. Contará con logueo de usuario y contraseña para ver y cambiar parámetros

5. Requerimientos opcionales (se llevarán a cabo si el tiempo es suficiente)

- 5.1. El nodo contará con una salida de dimerización de 220 VCA para ventilación
- 5.2. El servidor (Raspberry Pi) funcionará como *access point* al que se conectará el nodo

7. Historias de usuarios (*Product backlog*)

A continuación se enumeran las historias de usuario y se las pondera según la serie de Fibonacci. Los valores elegidos para la dificultad son 3, 5 y 8; para la complejidad 3, 5 y 8 y para la incertidumbre 2, 3 y 5.

- Como usuario deseo poder monitorear la temperatura en tiempo real de la habitación para saber si hace falta encender la calefacción.
Dificultad: 5 - Complejidad: 3 - Incertidumbre: 3 - Total: 11 - Valor más cercano 13
- Como usuario deseo que el sistema funcione igualmente sin internet para garantizar el confort en la sala.
Dificultad: 3 - Complejidad: 3 - Incertidumbre: 2 - Total: 8 - Valor más cercano 8
- Como usuario quiero que el sistema me de aviso por mail si la temperatura alcanza valores extremos para salvaguardar la integridad de los bienes.
Dificultad: 5 - Complejidad: 5 - Incertidumbre: 3 - Total: 13 - Valor más cercano 13
- Como usuario quiero programar la regulación de las luces en horarios programados para simular la presencia de personas en una casa que no está siempre habitada.
Dificultad: 5 - Complejidad: 8 - Incertidumbre: 3 - Total: 16 - Valor más cercano 21
- Como usuario quiero almacenar los valores sensados y los estados de las salidas para poder generar un posible ahorro de energía eléctrica.
Dificultad: 5 - Complejidad: 5 - Incertidumbre: 3 - Total: 13 - Valor más cercano 13
- Como usuario quiero poder controlar las salidas desde un mando dentro de la habitación para no depender de una PC o teléfono. Dificultad: 8 - Complejidad: 8 - Incertidumbre: 3 - Total: 19 - Valor más cercano 21

8. Entregables principales del proyecto

Los entregables del proyecto son:

- Diagrama de conexión del nodo
- Diagrama de circuito esquemático de accesorios del nodo
- Código fuente del nodo (ESP32)
- Código fuente del frontend y del backend
- Informe final

9. Desglose del trabajo en tareas

Las tareas se han agrupado por tema y ordenado por secuencialidad.

1. Investigación y búsqueda de material (45 hs)

- 1.1. Búsqueda y análisis de soluciones similares en el mercado (5 hs)
- 1.2. Búsqueda de códigos fuente de ejemplo similares (10 hs)
- 1.3. Búsqueda y análisis de los materiales a usar (5 hs)
- 1.4. Listado de materiales necesarios y aspectos de red (5 hs)
- 1.5. Estudio de funcionamiento básico de cada componente del sistema (10 hs)
- 1.6. Estudio y diseño de diagrama en bloques del sistema (5 hs)
- 1.7. Organizar la documentación y archivar lo importante (5 hs)
2. Adquisición de materiales (15 hs)
 - 2.1. Búsqueda de materiales (3 hs)
 - 2.2. Análisis de alternativas (2 hs)
 - 2.3. Armado y análisis de costos (5 hs)
 - 2.4. Negociación con proveedores y compra de materiales (5 hs)
3. Diseño y puesta en marcha de hardware (87 hs)
 - 3.1. Instalación de sistema operativo Linux en servidor (5 hs)
 - 3.2. Instalación y configuración de software adicional y dependencias (2 hs)
 - 3.3. Pruebas de funcionalidad básicas (5 h)
 - 3.4. Diseño de circuitos esquemático e impreso del hardware adicional al nodo (10 hs)
 - 3.5. Armado y pruebas del hardware adicional (20 hs)
 - 3.6. Conexión y pruebas del hardware adicional con ESP32 (40 hs)
 - 3.7. Documentar brevemente lo sucedido en este punto (5 hs)
4. Instalación y pruebas de la red local (23 hs)
 - 4.1. Instalación y configuración de la red local (10 hs)
 - 4.2. Pruebas iniciales de conexión del servidor a la red (5 h)
 - 4.3. Pruebas de conexión del nodo a la red (3 hs)
 - 4.4. Documentar brevemente lo sucedido en este punto (5 hs)
5. Desarrollo del software del servidor (frontend y backend) (302 hs)
 - 5.1. Instalación y configuración del software de la base de datos (5 hs)
 - 5.2. Instalación y configuración del servidor web (5 hs)
 - 5.3. Pruebas avanzadas de conexión a la red local e internet (2 hs)
 - 5.4. Creación del software para la verificación constante de conexión a internet (5 hs)
 - 5.5. Creación del software para conocer la hora y temperatura local por internet (5 hs)
 - 5.6. Creación del script para correr los programas automáticamente al encender el servidor (5 hs)
 - 5.7. Diseño e implementación de la base de datos (10 hs)
 - 5.8. Diseño y desarrollo de las funcionalidades del backend (40 hs)
 - 5.9. Pruebas de integración con la base de datos (10 hs)
 - 5.10. Desarrollo y configuración del código para enviar mails (20 hs)
 - 5.11. Diseño y maquetación de la página web (20 hs)

- 5.12. Desarrollo de la página web (40 hs)
- 5.13. Pruebas de funcionalidad de la página web (20 hs)
- 5.14. Integración de la página web con el software backend (30 hs)
- 5.15. Resolución de problemas de integración (20 hs)
- 5.16. Resolución de problemas de funcionamiento (40 hs)
- 5.17. Diseño de reportes a presentar por la aplicación (20 hs)
- 5.18. Documentar brevemente lo sucedido en este punto (5 hs)
- 6. Desarrollo del software de nodo (100 hs)
 - 6.1. Diseño de base de software de inicialización y configuración del nodo (5 hs)
 - 6.2. Diseño de software de lectura de entradas y actuación de salidas (20 hs)
 - 6.3. Diseño de software de comunicación con el servidor (30 hs)
 - 6.4. Pruebas de comunicación con el servidor (20 hs)
 - 6.5. Resolución de problemas (20 hs)
 - 6.6. Documentar brevemente lo sucedido en este punto (5 hs)
- 7. Instalación y pruebas funcionales (33 hs)
 - 7.1. Instalación del nodo y adicionales en la habitación (7 hs)
 - 7.2. Pruebas funcionales de los comandos locales dentro de la habitación (3 hs)
 - 7.3. Análisis y resolución de posibles fallas (3 hs)
 - 7.4. Pruebas reales de funcionalidad en un ambiente (10 hs)
 - 7.5. Pruebas de alarmas y funcionalidad de aplicación web (5 hs)
 - 7.6. Documentar brevemente lo sucedido en este punto (5 hs)

Cantidad total de horas: (605 hs).

Si en los cálculos anteriores llegara a haber sobrante de tiempo, se puede analizar diseñar e implementar el control de la salida de 220 VCA para controlar la ventilación. Dicha duración de tareas extra (opcionales) se estima de la siguiente manera:

- Diseño e implementación del control de ventilación mediante salida dimerizable (50 hs)
 - 1. Diseño de placa de control de la salida de 220 VCA (10 hs)
 - 2. Armado de placa para dimerizar salida de 220 VCA (10 hs)
 - 3. Conexión al nodo y pruebas (5 hs)
 - 4. Modificación del software del servidor (5 hs)
 - 5. Modificación del software del nodo (20 hs)

10. Diagrama de Activity On Node

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

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:



Figura 2. Diagrama en *Activity on Node*

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.

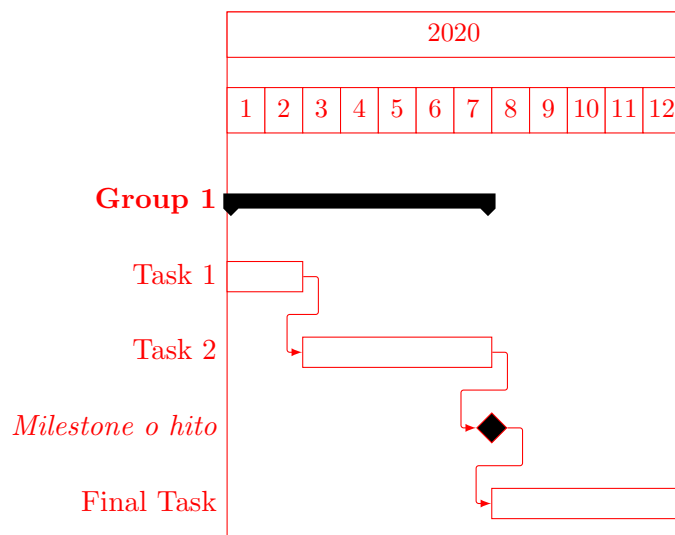


Figura 3. Diagrama de gantt de ejemplo



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):

■ Ocurrencia (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.