



Monitoreo y control de sistemas de bombeo de agua potable de pozos profundos

Autor:

Esp. Ing. Mario Fernando Aguilar Montoya

Director:

Mg. Ing. Mauricio Barroso Benavides (FIUBA)

Esta planificación fue realizada en el curso de Gestión de proyectos entre el 22 de junio de 2024 y el 17 de Agosto de 2024.

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	6
5. Supuestos del proyecto.	6
6. Requerimientos	7
7. Historias de usuarios (<i>Product backlog</i>).	7
8. Entregables principales del proyecto	9
9. Desglose del trabajo en tareas	9
10. Diagrama de Activity On Node.	10
11. Diagrama de Gantt	10
12. Presupuesto detallado del proyecto	14
13. Gestión de riesgos	14
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	22 de junio de 2024
1	Se completa hasta el punto 9 inclusive	2 de agosto de 2024

Acta de constitución del proyecto

Buenos Aires, 22 de junio de 2024

Por medio de la presente se acuerda con el Esp. Ing. Mario Fernando Aguilar Montoya que su Trabajo Final de la Maestría en Sistemas Embebidos se titulará “Monitoreo y control de sistemas de bombeo de agua potable de pozos profundos” y consistirá en esencialmente en la implementación de un prototipo de un sistema para el monitoreo y control de un sistema de bombeo de agua potable. El trabajo tendrá un presupuesto preliminar estimado de 680 hs de trabajo y un costo estimado de 46358 Bs, con fecha de inicio el 22 de junio de 2024 y fecha de presentación pública el 15 de mayo de 2025.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg
Director posgrado FIUBA

Ing. Carlos Alvarado Cruz
COSAAAL RL

Mg. Ing. Mauricio Barroso Benavides
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

En los últimos años la tendencia tecnológica se ha orientado hacia la creación de dispositivos electrónicos conectados a internet. Permitiendo la implementación de una amplia variedad de nuevas aplicaciones.

Cossalt RL es una cooperativa que da el servicio de agua a la ciudad de Tarija. Actualmente la cooperativa consta de 50 sistemas de bombeo de agua de pozos distribuidos por toda la ciudad para poder suministrar agua potable a las personas de cada zona. La mayoría de los sistemas son manuales esto quiere decir que hay un operario que se encarga de prender y apagar las bombas de agua dependiendo el nivel del agua en los tanques de almacenamiento todo de forma manual.

El objetivo principal del proyecto es la creación de un sistema capaz de realizar mediaciones de consumo eléctrico de las bombas, presión del agua en las tuberías, nivel del tanque de almacenamiento y enviar los datos obtenidos a una plataforma IoT haciendo uso de un módulo de comunicación GSM/GPRS. El sistema también controlara el encendido y apagado de las bombas de forma automática dependiendo las lecturas de los sensores. La plataforma IoT permitirá almacenar los datos, crear paneles de visualización y controlar actuadores. También se mostrarán los datos obtenidos en un display ubicado en el panel de control. En la figura 1 se puede observar el diagrama de bloques del sistema.

El proyecto permitirá al cliente disminuir el desperdicio de agua que ocurre al momento de bombear agua desde los pozos profundos a los tanques de almacenamiento, también ayudará a detectar de forma temprana posibles rupturas en las tuberías. Al tener un monitoreo y control del encendido y apagado de las bombas se disminuirá el consumo eléctrica.

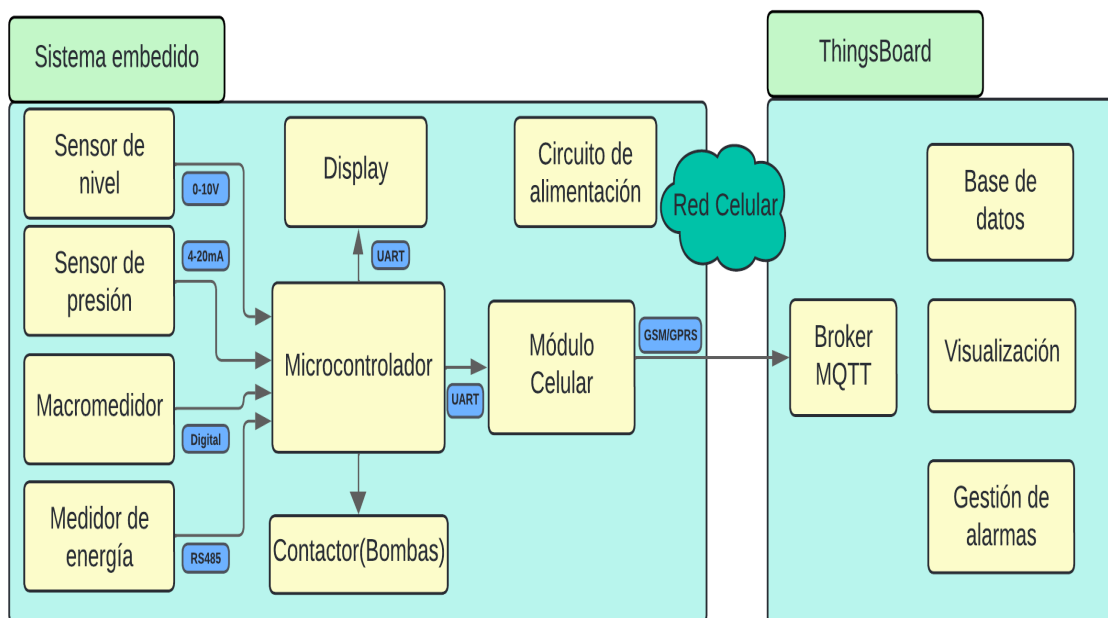


Figura 1. Diagrama en bloques del sistema.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Ing. Carlos Alvarado Cruz	COSAALT RL	Operador de sistema de bombeo
Responsable	Esp. Ing. Mario Fernando Aguilar Montoya	FIUBA	Alumno
Orientador	Mg. Ing. Mauricio Barroso Benavides	-	Director del Trabajo Final
Usuario final	Operarios	COSAALT RL	-

- Orientador: El Mg. Ing. Mauricio Barroso Benavides tiene mucha experiencia en el desarrollo de proyectos IoT, será una guía para el desarrollo del firmware y del hardware del embebido. Es exigente con los tiempos y la calidad del desarrollo del proyecto.

3. Propósito del proyecto

El propósito del proyecto es desarrollar un sistema que sea capaz de monitorear y controlar parámetros relevantes en un sistema de bombeo de agua, con la finalidad de disminuir el consumo energético de las bombas y el desperdicio de agua potable.

4. Alcance del proyecto

El proyecto incluye:

- Diseño e implementación del PCB, esto incluye todas las etapas desde el diseño de los esquemáticos hasta el envío para la fabricación y ensamblaje, finalmente las pruebas preliminares de funcionamiento.
- Desarrollo del firmware del microcontrolador basado en un sistema operativo de tiempo real.
- Configuración de la plataforma IoT para la visualización, almacenamiento, gestión de alarmas.

El proyecto no incluye:

- Diseño y fabricación del gabinete que aloja al dispositivo.
- Manuales de instalación y de usuario del dispositivo.

5. Supuestos del proyecto

- El tiempo de fabricación y ensamblaje de los PCBs de prueba estará dentro de lo planeado.

- El tiempo de importación de los módulos y componentes estarán dentro del tiempo esperado.
- El presupuesto no superará en gran medida lo estimado.
- No tener problemas de importación de los componentes.
- El tiempo disponible para la dedicación del proyecto será el adecuado para cumplir con los objetivos.

6. Requerimientos

1. Requerimientos funcionales:

1.1. Requerimientos de firmware

- 1) El firmware debe estar bajo un RTOS.
- 2) Se deberá llevar control de los cambios bajo el sistema de control de versiones Git.
- 3) El firmware debe poder suscribirse y publicar en tópicos de un broker MQTT.
- 4) El firmware debe comunicarse con el módulo GSM/GPRS mediante protocolo serial.
- 5) El firmware debe poder obtener las lecturas de los sensores.
- 6) El firmware debe poder actualizarse de forma remota.
- 7) Se deben realizar los drivers para los sensores.
- 8) Se debe hacer test unitarios para los drivers.

1.2. Requerimientos de hardware

- 1) El PCB tiene que tener un circuito de acondicionamiento para una entrada RS485.
- 2) El PCB tiene que tener un circuito de acondicionamiento para una entrada de 4-20mA.
- 3) El PCB tiene que tener un circuito de acondicionamiento para una entrada de 0-10V.
- 4) Debe contar con una display.

1.3. Requerimientos de interfaz gráfica

- 1) Debe mostrar los valores de los sensores.
- 2) Debe mostrar el estado de los actuadores.
- 3) Debe contar con botones para controlar la actuadores.
- 4) Debe almacenar los datos.
- 5) Debe poder establecer reglas y alarmas.

1.4. Requerimientos de documentación

- 1) Se debe presentar un informe de avance del proyecto.
- 2) Se debe presentar una memoria técnica al final del proyecto.

7. Historias de usuarios (*Product backlog*)

El criterio para asignar los puntos a las historias de usuario es el siguiente:

- Cantidad de trabajo a realizar
 - Bajo: peso 1
 - Medio: peso 3
 - Alto: peso 5
- Complejidad del trabajo a realizar
 - Bajo: peso 1
 - Medio: peso 3
 - Alto: peso 5
- Riesgo o incertidumbre del trabajo a realizar
 - Bajo: peso 1
 - Medio: peso 3
 - Alto: peso 5

Historia de usuario 1: “Como operador quiero ver en una pantalla local las lecturas de los sensores”

- Dificultad: medio(3)
- Complejidad: medio(3)
- Riesgo: bajo(1)

Story point=(3+3+1)=7

Historia de usuario 2: “Como operador quiero recibir alertas cuando sobrepase algún umbral en la lectura de los sensores”

- Dificultad: medio(3)
- Complejidad: alto(3)
- Riesgo: medio(3)

Story point=(3+3+3)=9

Historia de usuario 1: “Como operador quiero poder visualizar la información que genera el sistema en un dashboard que pueda ser accedido mediante internet “

- Dificultad: medio(3)
- Complejidad: medio(3)
- Riesgo: bajo(1)

Story point=(3+3+1)=7

8. Entregables principales del proyecto

- Documentación en formato pdf de los esquemáticos y planos del PCB.
- Código fuente del firmware no editable.
- Prototipo funcional.
- Documentación del proyecto.

9. Desglose del trabajo en tareas

1. Documentación del proyecto(80 hs)
 - 1.1. Escribir la planificación del proyecto(60 hs)
 - 1.2. Especificación de requisitos del firmware(10 hs)
 - 1.3. Definición de las pruebas de aceptación(10 hs)
2. Búsqueda de material bibliográfico (45 hs)
 - 2.1. Buscar hojas de datos de todos los componentes (15 hs)
 - 2.2. Estudiar el funcionamiento de cada uno de los componentes (15 hs)
 - 2.3. Investigar sobre dispositivos con funciones similares (15 hs)
3. Diseño del hardware del sistema (160 hs)
 - 3.1. Selección de componentes(15 hs)
 - 3.2. Diseño del esquemático(60 hs)
 - 3.3. Diseño de los símbolos(20 hs)
 - 3.4. Diseño del PCB(60 hs)
 - 3.5. Diseño de los componentes en 3D(5 hs)
4. Desarrollo del firmware(170)
 - 4.1. Diseño de la arquitectura del firmware(25 hs)
 - 4.2. Desarrollo de los drivers del firmware(70 hs)
 - 1) Desarrollo driver para el sensor de presión(20 hs)
 - 2) Desarrollo del driver para el sensor de nivel(20 hs)
 - 3) Desarrollo del driver para el macromedidor(10 hs)
 - 4) Desarrollo del driver para el medidor de energía(20 hs)
 - 4.3. Desarrollar el módulo de la aplicación(80 hs).
5. Configuración de la interfaz gráfica(70 hs)
 - 5.1. Diseño de la interfaz gráfica en la plataforma IoT(40 hs)
 - 5.2. Configuración de la base de datos, alarmas y reglas(30 hs)
6. Testing(30 hs)
 - 6.1. Testeo eléctrico del PCB(10 hs)

- 6.2. Testeo del firmware(10 hs)
- 6.3. Depuración del firmware(10 hs)
- 7. Cierre del proyecto(120 hs)
 - 7.1. Informes de avance del proyecto(20 hs)
 - 7.2. Elaboración de la memoria técnica del trabajo final(80 hs)
 - 7.3. Presentación final del proyecto(20 hs)

Cantidad total de horas: (680 hs)

10. Diagrama de Activity On Node

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

Una herramienta simple para desarrollar los diagramas es el Draw.io (<https://app.diagrams.net/>). Draw.io

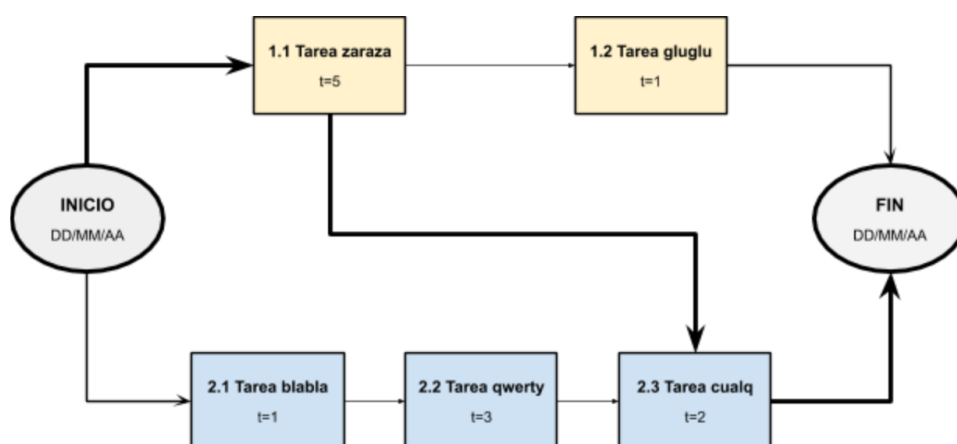


Figura 2. Diagrama de *Activity on Node*.

Indicar claramente en qué unidades están expresados los tiempos. De ser necesario indicar los caminos semi críticos y analizar sus tiempos mediante un cuadro. Es recomendable usar colores y un cuadro indicativo describiendo qué representa cada color.

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.

Las fechas pueden ser calculadas utilizando alguna de las herramientas antes citadas. Sin embargo, el siguiente ejemplo fue elaborado utilizando [esta hoja de cálculo](#).

Es importante destacar que el ancho del diagrama estará dado por la longitud del texto utilizado para las tareas (Ejemplo: tarea 1, tarea 2, etcétera) y el valor $x unit$. Para mejorar la apariencia del diagrama, es necesario ajustar este valor y, quizás, acortar los nombres de las tareas.

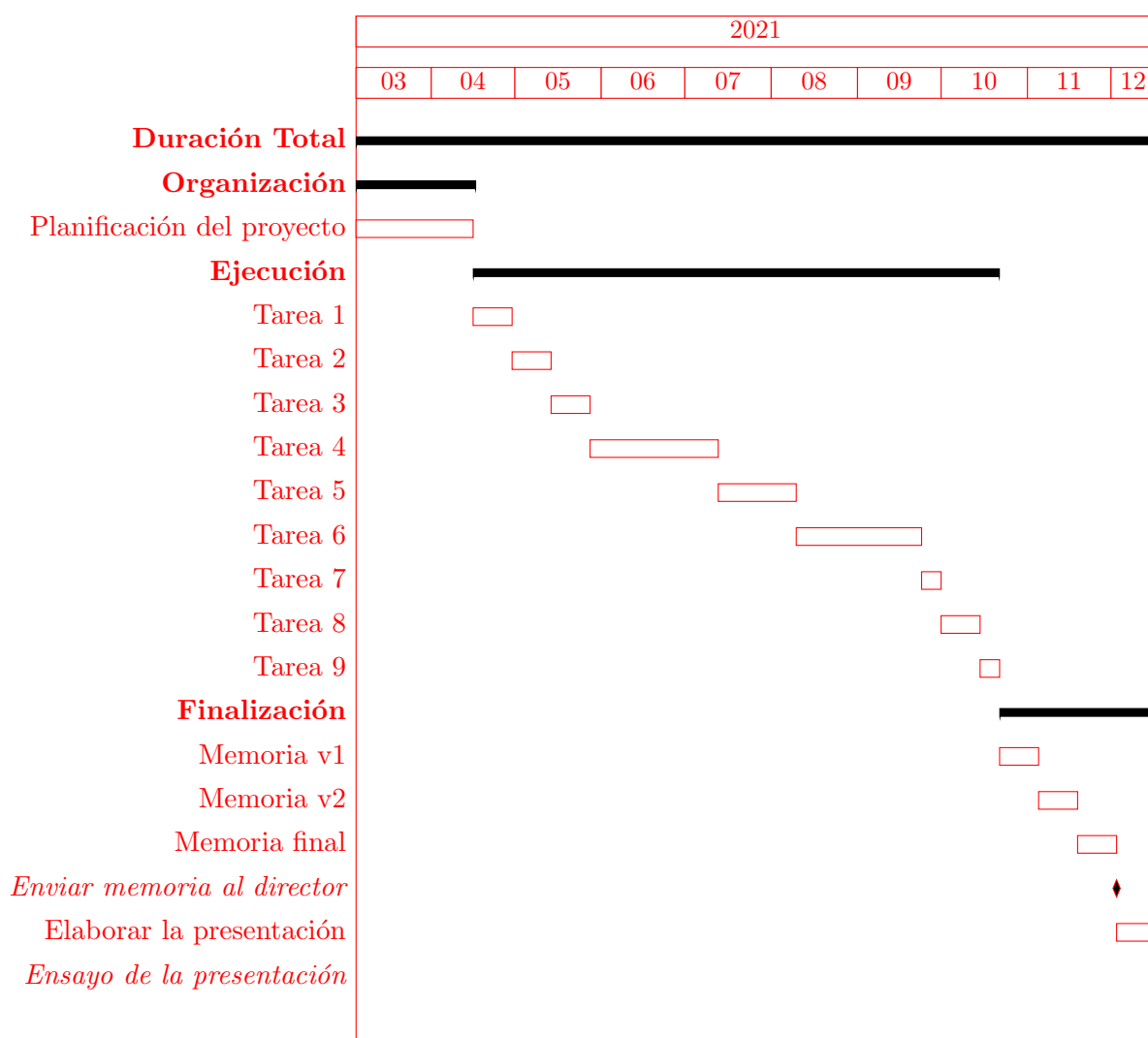


Figura 3. Diagrama de gantt de ejemplo



Figura 4. Ejemplo de diagrama de Gantt (apaisado).

12. Presupuesto detallado del proyecto

Los precios expresados en la siguiente tabla se encuentran en Bolivianos Bs.

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Tarjeta de desarrollo	1	150	150
Módulo de comunicación GSM/GPRS	1	350	350
Componentes electrónicos para el PCB	1	400	400
Fabricación y emsamblaje del PCB	5	50	250
Pantalla nextion	1	260	260
Caja de la tarjeta PCB	1	250	250
Horas de ingeniería	680	50	34000
SUBTOTAL			35660
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
30 % de los costos directos	1	1	10698
SUBTOTAL			10698
TOTAL			46358

13. Gestión de riesgos

a) Identificación de los riesgos y estimación de sus consecuencias:

Riesgo 1: Detección de errores en el esquemático o el ruteo del PCB ya habiendo fabricado los prototipos.

- Severidad (7): Se retrasa el proyecto por unas semanas hasta volver a mandar a fabricar los PCBs nuevamente.
- Probabilidad de ocurrencia (6): Dada la moderada experiencia del alumno es posible que ocurra.

Riesgo 2: Mala estimación de la planificación.

- Severidad (8): El proyecto sufrirá varios cambios y retrasos en la ejecución.
- Probabilidad de ocurrencia (6): No cuenta con mucha experiencia en planificación de proyectos IoT.

Riesgo 3: Retraso en la programación del firmware.

- Severidad (7): Retrasos en el proyecto debido a que el firmware es una parte fundamental en el momento de la integración e implementación del prototipo.
- Probabilidad de ocurrencia (5): Falta de conocimiento de la arquitectura del microcontrolador.

Riesgo 4: Retraso en la fabricación y ensamblaje de los PCBs.

- Severidad (7): Por falta del hardware no se podrá probar el firmware sobre el hardware del prototipo.
- Probabilidad de ocurrencia (4): Se mandara a fabricar y ensamblar a una empresa de china.

Riesgo 5: Pérdida o daño de los archivos asociados al proyecto.

- Severidad (7): Al perder los archivos del proyecto se tendra que volver reiniciar disenos.
- Probabilidad de ocurrencia (3): Se llevara un control de versiones con los documentos del proyecto.

b) Tabla de gestión de riesgos:

Riesgo	S	O	RPN	S*	O*	RPN*
Detección de errores en el esquemático o el ruteo del PCB ya habiendo fabricado los prototipos	7	6	42	7	5	35
Mala estimación de la planificación	8	6	48	8	4	32
Retraso en la programación del firmware	7	5	35			
Retraso en la fabricación y ensamblaje de los PCBs	7	4	28			
Pérdida o daño de los archivos asociados al proyecto	7	3	21			

Criterio adoptado:

Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 42

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*: Se revisara minuciosamente los esquematicos, el ruteo del PCB, y se buscara ayuda en algun ingeniero que tenga experiencia en desarrollo de circuitos impresos en proyectos de estas características.

- Severidad (7):No cambia.
- Probabilidad de ocurrencia (5): Menor, alcontar con ayuda de profesionales expertos en el tema.

Riesgo 2*: Se consultara con el director en el momento de realizar la planificación para ajustar los tiempos de las tareas de acuerdo a su experiencia en este tipo de proyectos.

- Severidad (8):No cambia.
- Probabilidad de ocurrencia (4): Se reduce el riesgo de hacer una mala planificación.

14. Gestión de la calidad

Para cada uno de los requerimiento funcionales, a constitución, se lista su respectiva validacion y verificación. Las mismas llevadas a cabo por el responsable del proeyecto.

- Req #1: El firmware debe estar bajo un RTOS.
 - Verificación: .
 - Validación: .
- Req #2: Se deberá llevar control de los cambios bajo el sistema de control de versiones Git.
 - Verificación: Utilizar una consola para mandar comandos git.
 - Validación: Mostrar los commits realizados durante el proyecto.
- Req #3: El firmware debe poder suscribirse y publicar en tópicos de un broker MQTT.
 - Verificación: Publicar datos a un topico del borker MQTT.
 - Validación: Visualizar los datos en la interfaz grafica en la plataforma IoT.
- Req #4: El firmware debe comunicarse con el módulo GSM/GPRS mediante protocolo serial.
 - Verificación: Mandar comandos AT por puerto serial y esperar la respuesta del modulo.
 - Validación: Visualizar las respuesta del modulo en un monitor serial.
- Req #5: Se deben realizar los drivers para los sensores.
 - Verificación: Hacer un test unitario para cada driver.
 - Validación: Mostrar los resultados de covertura en GCOV.
- Req #6: El PCB tiene que tener un circuito de acondicionamiento para una entrada RS485.
 - Verificación:.
 - Validación:.
- Req #7: Debe contar con una display.
 - Verificación: Mandar datos al display para visualizarlos.
 - Validación: Mostrar lecturas de los sensores en graficas en el display.
- Req #8: Debe mostrar los valores de los sensores.
 - Verificación: Mandar datos leidos de los sensores a un topico por MQTT a la plataforma.
 - Validación: Entrar a la plataforma IoT y visualizar los datos.
- Req #9: Debe contar con botones para controlar la actuadores.
 - Verificación: Visualizar los datos recibidos en un monitor serial al momento de presionar un boton en la plataforma IoT.

- Validación: Oprimir un boton en la interfaz y visualizar la activacion de un actuador.
- Req #10: Requerimientos de documentación.
 - Verificación: Verificar que se haya presentado el informe de avance y la memoria tecnica del proyecto.
 - Validación: Revisar que se cumplio los requerimientos estipulados en el plan de proyectos.

Elija al menos diez requerimientos que a su criterio sean los más importantes/críticos/que aportan más valor y para cada uno de ellos indique las acciones de verificación y validación que permitan asegurar su cumplimiento.

- Req #1: copiar acá el requerimiento con su correspondiente número.
 - 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

- Pautas de trabajo que se seguirán para analizar si se respetó el Plan de Proyecto original:
 - Encargado: Ing. Mario Aguilar Montoya
 - Procedimiento: Durante el proceso de cierre se evaluara si se han cumplido los plazos de entrega y ejecucion de cada tarea planteada en el Plan de Proyecto. En el caso de hallar requerimientos incumplidos y/o retrasos en las tareas se evaluaran las causa y se propondran acciones para evitarlo en futuros proyectos.
- Identificación de las técnicas y procedimientos útiles e inútiles que se emplearon, los problemas que surgieron y cómo se solucionaron:
 - Encargado: Ing. Mario Aguilar Montoya
 - Procedimiento: Se evaluaran los procedimientos utilizados en funcion a su utilidad y eficiencia para alcanzar los objetivos predefinidos.
- Acto de cierre:
 - Encargado: Ing. Mario Aguilar Montoya
 - Procedimiento: Se realizara la defensa publica del trabajo final ante el jurado, posterior a esto se procedera a agradecer a todos los que ayudaron a realizar el proyecto, director del trabajo, miembros del jurado y autoridades de la CESE.