



Diseño e implementación de un data logger para mediciones meteorológicas

Autor:

Ing. Renato Barresi

Director:

Esp. Lic. Pablo Zizzutti (FIUBA)

Esta planificación fue realizada en el curso de Gestión de proyectos entre el 01 de marzo de 2022 y el 19 de abril 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	6
5. Supuestos del proyecto.	7
6. Requerimientos	8
7. Historias de usuarios (<i>Product backlog</i>).	8
8. Entregables principales del proyecto	10
9. Desglose del trabajo en tareas	10
10. Diagrama de Activity On Node.	11
11. Diagrama de Gantt	12
12. Presupuesto detallado del proyecto	14
13. Gestión de riesgos	14
14. Gestión de la calidad	16
15. Procesos de cierre	18

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	01 de marzo de 2022
1	Se completa hasta el punto 5 inclusive	15/03/2022
2	Se completa hasta el punto 9 inclusive y agregan correcciones	27/03/2022

Acta de constitución del proyecto

Buenos Aires, 01 de marzo de 2022

Por medio de la presente se acuerda con el Ing. Ing. Renato Barresi que su Trabajo Final de la Carrera de Especialización en Sistemas Embebidos se titulará “Diseño e implementación de un data logger para mediciones meteorológicas”, consistirá esencialmente en la implementación de un prototipo de computador que hará de interfaz con sensores meteorológicos y el internet, y tendrá un presupuesto preliminar estimado de 600 hs de trabajo y \$ 1000 US, con fecha de inicio 01 de marzo de 2022 y fecha de presentación pública 15 de mayo de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg
Director posgrado FIUBA

Javier Marin
Tech Enterprise S.A

Esp. Lic. Pablo Zizzutti
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

El cambio climático es una de las mayores problemáticas que enfrenta la humanidad en el siglo 21, se estima que 800 millones de personas son vulnerables a los efectos negativos que puede generar este, es por eso que se deben generar políticas enfocadas a luchar y contrarrestar los perjuicios que pueden ser causados por el cambio climático. Estas políticas deben estar basadas en datos y mediciones científicas, las cuales se obtienen en gran parte por medio de estaciones meteorológicas automáticas.

Una estación meteorológica automática se encarga de tomar mediciones meteorológicas, procesarlas, guardarlas y transmitirlos. El elemento principal de esta es un data logger, que es un computador con distintos periféricos para cumplir las funciones requeridas por la estación.

En Paraguay, el mercado de estaciones meteorológicas automáticas esta dominado por productos del extranjero, las que tienen costo muy elevado para el presupuesto que manejan las entidades publicas y privadas, también, el bajo nivel de customización que ofrecen, es un impedimento para poder modificar a la estación en base a las posibilidades del cliente.

Actualmente, Tech Enterprise S.A cuenta con data loggers propios, basados en microcontroladores de 8 bits, como el atmega328p y atmega2560. La baja cantidad de periféricos, velocidad, memoria RAM y FLASH con los que cuentan estos microcontroladores, limita las funcionalidades y escalabilidad del código del data logger. Es por esto que la empresa busca migrar a microcontroladores de 32 bits, como el STM32F767ZI, que cuenta con una frecuencia máxima de 216 MHz, memoria FLASH de 2 Mbytes y memoria SRAM de 512 Kbytes. También, cuenta con interfaz MAC que permite utilizar integrados PHY para Ethernet.

En lo que respecta al presente trabajo, se propone diseñar una solución basada en el mencionado microcontrolador, que sea capaz de comunicarse con sensores utilizando el protocolo de comunicación SDI-12, con una memoria FLASH y memoria SD y que pueda transmitir la información por medio de protocolos de capa 2 como el PPP y Ethernet.

En la figura 1 se muestra el diagrama en bloques del sistema. El microcontrolador se encarga de comunicarse con distintos periféricos, como la memoria SD, memoria Flash, ADC, etc. Además de poder comunicarse con dichos componentes, debe de ser capaz de actuar como cliente FTP.

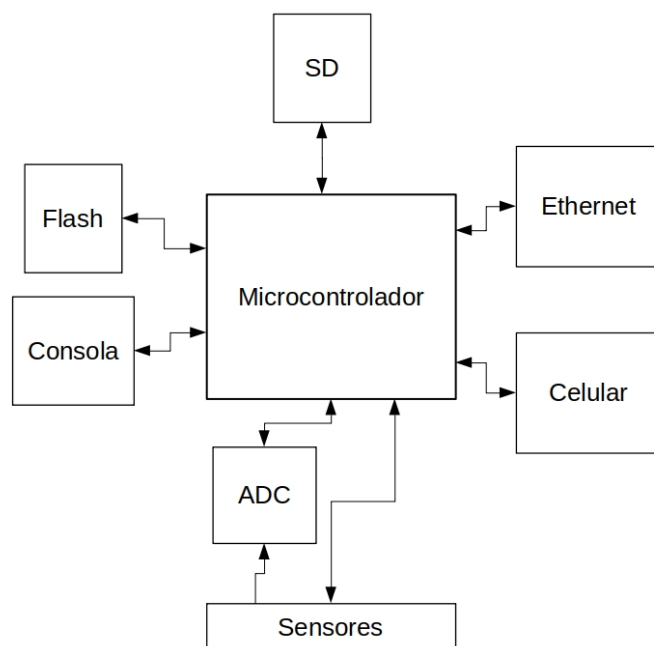


Figura 1. Diagrama en bloques del sistema.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Auspiciante	Jose Flecha	Data system S.A	Director ejecutivo
Cliente	Javier Marin	Tech Enterprise S.A	Director ejecutivo
Impulsor	Julio Mendez	CONACYT	Director PROINOVA
Responsable	Ing. Renato Barresi	FIUBA	Alumno
Orientador	Esp. Lic. Pablo Zizzutti	FIUBA	Director Trabajo final
Opositores	Jack Johnson	Campbell Scientific	Sales manager
Usuario final	Fernando Pio Barrios	Dirección de Meteorología e Hidrología	Director

3. Propósito del proyecto

El propósito de este proyecto es diseñar e implementar el prototipo de un data logger basado en un microcontrolador de 32 bits que permita comunicarse con sensores meteorológicos, procesar los datos, almacenar la información a una memoria SD y transmitir a un servidor FTP.

4. Alcance del proyecto

El proyecto incluye los siguientes puntos:

- Diseño, programación e implementación de un sistema embebido que sea capaz de comunicarse con distintos sensores, procesar los datos, guardar y transmitirlos por internet.
- Diseño de una biblioteca modular en C, que permita comunicarse con sensores que utilizan el protocolo SDI-12.
- Diseño de una biblioteca modular en C, que permita comunicarse con memorias de tipo FLASH.
- Diseño de la arquitectura del sistema embebido.
- Prototipo en protoboard del sistema embebido.
- Diseño de la placa de circuito impreso.
- Verificación funcional de la placa de circuito impreso.

Queda excluido del alcance del proyecto:

- Desarrollo de un cargador solar.
- Diseño de un RTOS.
- Diseño de un stack TCP/IP.
- Testeo de más de una semana del sistema.
- Testeo de rangos de temperatura y humedad mínimos y máximos.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- Se contará con una placa NUCLEO-F767ZI.
- Se podrá contar con los insumos necesarios en tiempo y forma.
- Se contará con las instalaciones disponibles para las pruebas pertinentes.
- Se dispondrá del tiempo necesario para el desarrollo del proyecto.
- Se contará con sensores meteorológicos para realizar las mediciones.
- Se contará con un módem celular para conexiones GPRS/GSM.
- Se cuenta con toda la información técnica de sensores meteorológicos, módem, microcontroladores, Ethernet, etc.

6. Requerimientos

1. Requerimientos de firmware.
 - 1.1. El proyecto sera desarrollado y mantenido mediante el control de versiones GIT.
 - 1.2. El proyecto sera desarrollado en lenguaje C, compilado con ARM GCC.
 - 1.3. Se deberá de desarrollar biblioteca de control del conversor analógico digital.
 - 1.4. Se deberá de desarrollar biblioteca de control de la memoria FLASH.
 - 1.5. Se deberá de desarrollar biblioteca del protocolo SDI-12.
2. Requerimientos de documentación.
 - 2.1. El código será documentado con la herramienta Doxygen.
 - 2.2. El ciclo de vida del proyecto sera respaldado por una memoria técnica.
 - 2.3. Se entregaran informes de avance.
3. Requerimientos asociados con la arquitectura del sistema.
 - 3.1. El data logger se debe basar en el STM32F767ZI.
 - 3.2. El STM32F767ZI se debe comunicar con una memoria FLASH por medio del protocolo SPI.
 - 3.3. El STM32F767ZI se debe comunicar con una memoria SD, por medio del protocolo SPI.
 - 3.4. El STM32F767ZI se debe comunicar con un conversor analógico digital, por medio del protocolo I2C.
4. Requerimientos de la interfaz.
 - 4.1. El sistema debe contar con protección contra cortocircuitos, sobre voltaje y bajo voltaje.
 - 4.2. El sistema debe funcionar con un rango de tension entre 7V y 24 VDC.
 - 4.3. El sistema debe contar con dos puertos RS-232.
 - 4.4. El sistema debe contar con un puerto Ethernet.
 - 4.5. El sistema debe contar con una interfaz SD.
 - 4.6. El sistema debe contar con un puerto micro USB.
 - 4.7. El sistema debe contar con borneras para conexión con los sensores.

7. Historias de usuarios (*Product backlog*)

Roles:

- Usuario/a - Persona que hará uso de la información proveniente de la estación.
- Instalador/a - Es la persona encargada del ensamblaje y puesta en marcha de la estación de Tech Enterprise S.A. A su cargo se encuentra el diseño y ensamblado de la estación para el usuario, esto es, elegir los sensores en función a las necesidades del usuario, tipo de alimentación, forma de conexión con la red, etc.

- Cliente - Es la persona que auspiciara el proyecto de la estación.

Ponderación:

- Este puntaje esta dado por la relevancia que tiene la historia de usuario para el éxito del proyecto definido en esta planificación.
- Se utilizará una escala de prioridad de 1, 2, y 4 puntos para las historias.
 - Dificultad/tiempo.
 - Complejidad
 - Incertidumbre/Riesgo.

Historias de usuario:

- Como ingeniero agrónomo, quiero poder utilizar los sensores que utilicen el protocolo SDI-12.
 - Dificultad/tiempo: 2.
 - Complejidad: 2.
 - Incertidumbre/Riesgo: 1.
 - Total: 5 pts.
- Como instalador, deseo poder actualizar el software de la estación de forma remota, subiendo el archivo binario al servidor FTP (4p).
 - Dificultad/tiempo: 4.
 - Complejidad: 4.
 - Incertidumbre/Riesgo: 4.
 - Total: 12 pts.
- Como meteorólogo, quiero poder recibir la información en un archivo .txt (4p).
 - Dificultad/tiempo: 1.
 - Complejidad: 1.
 - Incertidumbre/Riesgo: 1.
 - Total: 3 pts.
- Como project mánager deseo poder utilizar el protocolo Ethernet para realizar conexiones a internet.
 - Dificultad/tiempo: 2.
 - Complejidad: 4.
 - Incertidumbre/Riesgo: 1.
 - Total: 7 pts.

8. Entregables principales del proyecto

- Informe final.
- Manual de configuración y puesta en marcha.
- Diagrama de circuitos esquemáticos.
- Código fuente del firmware.
- Informe de avance.

9. Desglose del trabajo en tareas

1. Planificación del proyecto (40hs)
 - 1.1. Realizar el plan de proyecto (20 hs).
 - 1.2. Redactar requisitos del firmware (20 hs).
2. Investigación preliminar (40 hs)
 - 2.1. Investigar características técnicas de cada uno de los componentes a utilizar (16 hs).
 - 2.2. Investigar bibliotecas TCP/IP (12 hs).
 - 2.3. Investigar sobre sistemas operativos de tiempo real (12 hs).
3. Desarrollo de firmware (300 hs)
 - 3.1. Configuración inicial del microcontrolador y sus distintos periféricos (20 hs).
 - 3.2. Diseñar algoritmo de configuración de la estación (30 hs).
 - 3.3. Diseñar biblioteca para control de memoria FLASH (40 hs).
 - 3.4. Integrar biblioteca para control de memoria SD (10 hs).
 - 3.5. Diseñar biblioteca para comunicación por medio del protocolo SDI-12 (40 hs).
 - 3.6. Diseñar biblioteca para control del conversor analógico digital (40 hs).
 - 3.7. Diseñar biblioteca para cliente FTP (40 hs).
 - 3.8. Integración de las distintas bibliotecas (40 hs).
 - 3.9. Diseñar máquina de estados finitos (40 hs).
4. Validación (80 hs).
 - 4.1. Ejecutar esquema de pruebas de conexión PPP (20 hs).
 - 4.2. Ejecutar esquema de pruebas de conexión Ethernet (20 hs).
 - 4.3. Ejecutar esquema de pruebas de guardado en memoria SD (10 hs).
 - 4.4. Ejecutar esquema de pruebas de protocolo SDI-12 (5 hs).
 - 4.5. Ejecutar esquema de pruebas de cliente FTP (25 hs).
5. Documentación (30 hs).
 - 5.1. Documentación HTML con Doxygen (5 hs).
 - 5.2. Redacción de informes de avance (25 hs).

6. Elaboración final (110 hs).

6.1. Redacción del manuscrito final (40 hs).

6.2. Elaboración de la memoria técnica (40 hs).

6.3. Preparación de la exposición del trabajo final (20 hs).

6.4. Ensayo de la exposición (10 hs).

Cantidad total de horas: (600 hs).

10. Diagrama de Activity On Node

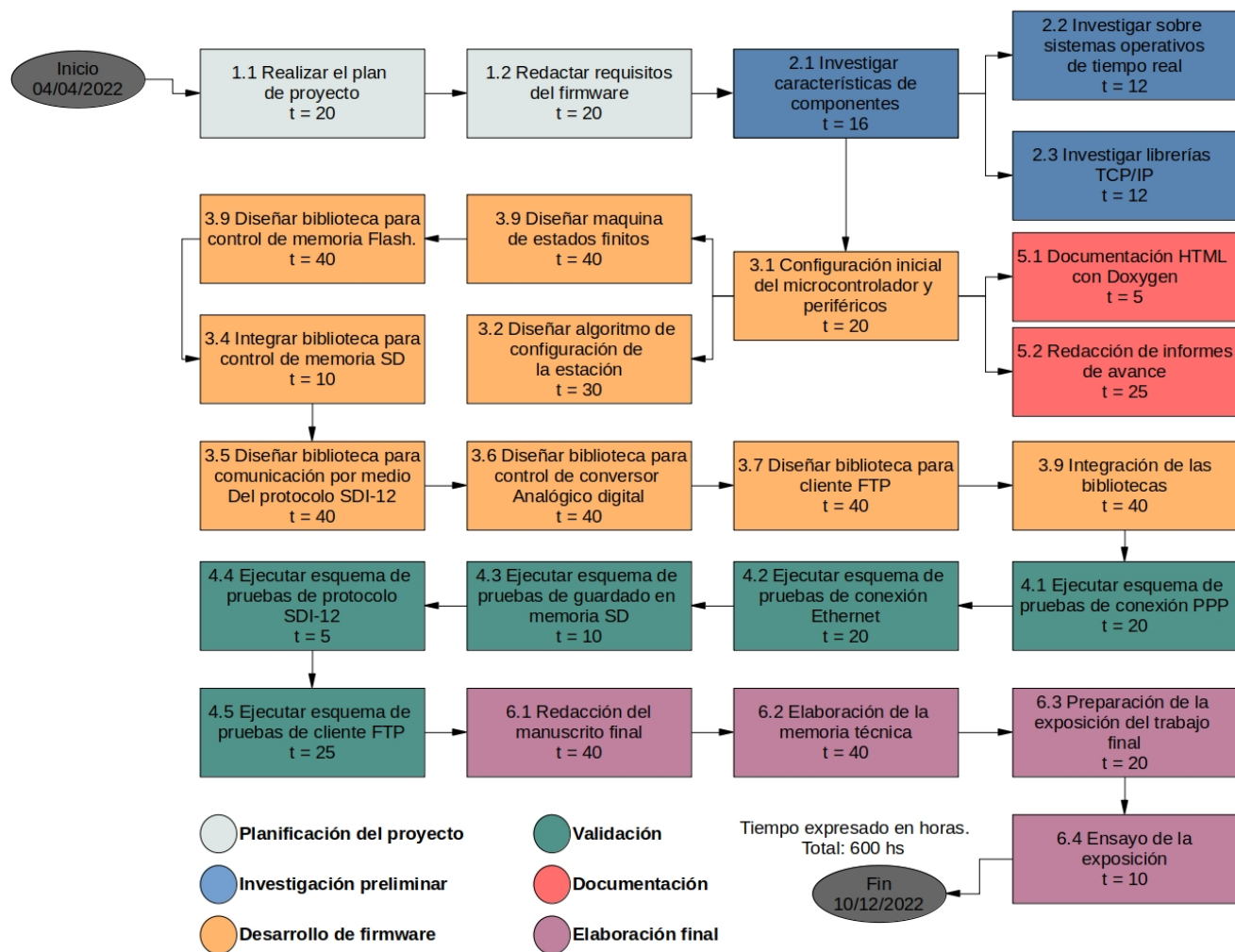


Figura 2. Diagrama en *Activity on Node*

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.

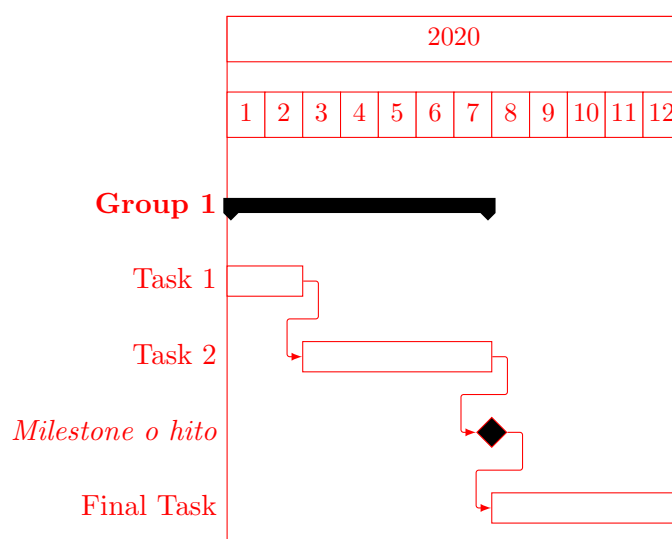


Figura 3. Diagrama de gantt de ejemplo



Figura 4. Ejemplo de diagrama de Gantt rotado

12. Presupuesto detallado del proyecto

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Mano de obra	600	\$ 5	\$ 3000
Placa Nucleo	1	\$ 60	\$ 60
Modem celular	1	\$ 90	\$ 90
Memoria SD	1	\$ 15	\$ 15
Memoria Flash	1	\$ 5	\$ 5
Conversor analogico digital	1	\$ 20	\$ 20
Sensor de suelo	1	\$ 200	\$ 200
Sensor de corriente	1	\$ 24	\$ 24
SUBTOTAL			\$ 3414
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Insumos varios	1	\$ 3000	\$ 3000
SUBTOTAL			\$ 3000
TOTAL			\$ 6414

13. Gestión de riesgos

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

Riesgo 1: La placa nucleo-F767ZI deja de funcionar.

- Severidad (S): 8
La severidad es moderada/alta, debido a que si la placa deja de funcionar demoraría en el inicio de la implementación del firmware.
- Ocurrencia (O): 2
La probabilidad es baja, ademas se cuenta con una placa de repuesto.

Riesgo 2: No se cuenta con un módem celular.

- Severidad (S): 5
La severidad es moderada, debido a que se requiere a un módem celular para poder desarrollar la implementación de los módulos dependientes de este.
- Ocurrencia (O): 1
La probabilidad es baja, debido a que se cuenta con disponibilidad del modem.

Riesgo 3: Imposibilidad de cumplir los plazos planteados.

- Severidad (S): 7
La severidad es moderada, algunos requisitos podrían requerir tiempos mas amplios de implementación que los presentados en el proyecto.
- Ocurrencia (O): 4
La probabilidad es moderada/baja, debido a que existe una planificación de previa.

Riesgo 4: Escasez del hardware en el mercado.

- Severidad (S): 8
La severidad es alta, esto seria un impedimento para el desarrollo de distintos módulos del proyecto.
- Ocurrencia (O): 4
La probabilidad es moderada/baja, debido a que el hardware seleccionado es accesible en el mercado.

Riesgo 5: Conexión al internet es de baja calidad.

- Severidad (S): 6
La severidad es moderada, debido a que es un requerimiento principal del sistema.
- Ocurrencia (O): 4
La probabilidad es moderada/baja, debido a que el stack a utilizar esta bien establecido.

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

Riesgo	S	O	RPN	S*	O*	RPN*
La placa nucleo-F767ZI deja de funcionar	8	2	16			
No se cuenta con un modem celular	5	1	5			
Imposibilidad de cumplir los plazos planteados	7	4	28	6	3	18
Escasez del hardware en el mercado	8	4	32	7	2	14
Conexión al internet es de baja calidad	6	4	24	5	2	10

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 20

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 3: Imposibilidad de cumplir los plazos planteados

- Plan de mitigación: Re asignación de horas atrasadas y rigurosidad con el cumplimiento.
- Severidad (6): La severidad es moderada, algunos requisitos podrían requerir tiempos mas amplios de implementación que los presentados en el proyecto.
- Ocurrencia (3): La probabilidad es baja, debido a que existe una planificación de previa, y rigurosidad con el cumplimiento de estas.

Riesgo 4: Escasez del hardware en el mercado

- Plan de mitigación: Compra de componentes lo antes posible.
- Severidad (7): La severidad es moderada, esto seria un impedimento para el desarrollo de distintos módulos del proyecto.

- Ocurrencia (2): La probabilidad es baja, debido a que el hardware seleccionado es accesible en el mercado y se comprara con anticipación a los componentes necesarios.

Riesgo 5: Conexión al internet es de baja calidad

- Plan de mitigación: Se realizaran los estudios correspondientes con antelación para la correcta configuración del stack.
- Severidad (5): La severidad es moderada, debido a que es un requerimiento principal del sistema.
- Ocurrencia (2): La probabilidad es baja, debido a que el stack a utilizar esta bien establecido y se estudiará con tiempo la documentación.

14. Gestión de la calidad

- Requerimiento 1.1: Se deberá de poder denir la fecha y horario del sistema, utilizando el RTC.
 - Verificación: Se implementara un menu de configuracion donde el usuario podra configurar al RTC.
 - Validación: Se leera el horario del RTC.
- Requerimiento 1.2: Se deberá de poder denir el periodo de transmisión de los archivos al servidor.
 - Verificación: Se implementara un menu de configuracion donde el usuario podra definir el periodo de transmision.
 - Validación: Se verificara si la transmision se realiza en el periodo establecido.
- Requerimiento 1.3: Se deberá de poder denir el periodo de la toma de muestras de los sensores.
 - Verificación: Se implementara un menu de configuracion donde el usuario podra definir el periodo de muestreo de los sensores.
 - Validación: La comunicacion con los sensores debe de ser la que el usuario definio.
- Requerimiento 1.4: Se deberá de poder defir el nombre particular de la estación.
 - Verificación: Se implementara un menu de configuracion donde el usuario podra definir el nombre de la estacion.
 - Validación: El nombre de los archivos enviados al servidor deben de contener el nombre definido por el usuario.
- Requerimiento 1.5: Se deberá de poder seleccionar los sensores conectados.
 - Verificación: Se implementara un menu de configuracion donde el usuario podra definir que sensores estan conectados al data logger.
 - Validación: El data logger debera de comunicarse con los sensores seleccionados por el usuario.
- Requerimiento 1.6: Se podrá denir el tipo de conexión para la red.

- Verificación: Se implementara un menu de configuracion donde el usuario podra definir que tipo de conexion de red cuenta el data logger.
- Validación: La conexion a internet debe de darse por medio de la conexion definida por el usuario.
- Requerimiento 2.1: El software deberá de poder comunicarse con sensores que utilicen el protocolo SDI-12.
 - Verificación: Se implementara una biblioteca para el manejo de comunicaciones por el protocolo SDI-12
 - Validación: El data logger debera de ser capaz de obtener datos de un sensor SDI-12.
- Requerimiento 2.2: El software deberá de poder adquirir datos de sensores analógicos.
 - Verificación: Se implementara una biblioteca para el manejo de comunicacion con el ADS1115 por medio del protocolo I2C.
 - Validación: El microcontrolador debera de ser capaz de comunicarse con el ADS1115 y obtener datos de un sensor de corriente.
- Requerimiento 2.3: El software deberá de poder poner al microcontrolador entrar en modo de bajo consumo.
 - Verificación: Se implementara un modulo que permita entrar y salir al microcontrolador de modo de bajo consumo.
 - Validación: Luego de realizar la transmision al servidor, el microcontrolador debe de entrar en modo de bajo consumo.
- Requerimiento 3.1: El software podrá detectar si la memoria SD esta conectada o no.
 - Verificación: Se implementara la logica de deteccion de la conexion con la memoria SD.
 - Validación: Se verificara si el software no realiza operaciones con la memoria SD si la memoria no esta conectada, y viceversa.
- Requerimiento 3.2: El software podrá escribir en la memoria SD archivos de tipo .txt
 - Verificación: Se utilizara la biblioteca FatFS para la operacion con archivos de texto y el protocolo SPI para la comunicacion con la memoria SD.
 - Validación: Se leera la informacion contenida en la memoria SD en una computadora personal.
- Requerimiento 4.1: El software deberá de poder transmitir archivos de tipo .txt a un servidor FTP.
 - Verificación: Se implementara una biblioteca de cliente FTP.
 - Validación: Se verificara en el servidor FTP si el software escribio el archivo.
- Requerimiento 4.2: El software deberá de poder parsear archivos .txt guardados en un servidor FTP.
 - Verificación: Se implementara una funcion que podra descargar archivos .txt del servidor FTP y parsear la informacion.
 - Validación: Se verificara que el microcontrolador actue en funcion a la informacion parseada.

- Requerimiento 4.3: El software deberá de poder enviar mensajes de diagnostico a un servidor TCP.
 - Verificación: Se implementara una biblioteca para el cliente TCP.
 - Validación: Se verificara que los mensajes TCP enviados por el software lleguen a un servidor TCP montado.

15. Procesos de cierre

- Pautas de trabajo que se seguirán para analizar si se respetó el Plan de Proyecto original:
 - Responsable: Renato Barresi.
 - Actividad: Se compararán los tiempos de ejecución con los planificados.
- Identificación de las técnicas y procedimientos útiles e inútiles que se emplearon, y los problemas que surgieron y cómo se solucionaron:
 - Responsable: Renato Barresi.
 - Actividad:
 - Se evaluará a los diferentes procedimientos y técnicas utilizadas para evaluar el impacto en el proyecto.
 - Se documentarán los problemas encontrados con sus respectivas soluciones.
- Indicar quién organizará el acto de agradecimiento a todos los interesados, y en especial al equipo de trabajo y colaboradores:
 - Responsable: Renato Barresi.
 - Actividad: Al finalizar el proyecto y la presentación publica, se procederá a agradecer a todas las personas que formaron parte del desarrollo del proyecto, docentes, jurados, compañeros y autoridades de la carrera.