



Desarrollo de aplicación Hospitalaria con MQTT

Autor:

Ing. Gustavo Bastian

Director:

Mg.Ing. Ericson Joseph Estupiñan Pineda (Surix S.R.L)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 31 de Octubre de 2021 y el 7 de Diciembre de 2021.*

Índice

1. Descripción técnica-conceptual del proyecto a realizar	5
2. Identificación y análisis de los interesados	7
3. Propósito del proyecto	7
4. Alcance del proyecto	7
5. Supuestos del proyecto.	8
6. Requerimientos	8
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	15
12. Presupuesto detallado del proyecto	19
13. Gestión de riesgos	19
14. Gestión de la calidad	20
15. Procesos de cierre	20

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	31/10/2021
1	Se completa hasta el punto 5 inclusive	3/11/2021
2	Se completa hasta el punto 9 inclusive	11/11/2021

Acta de constitución del proyecto

Buenos Aires, 31 de Octubre de 2021

Por medio de la presente se acuerda con el Ing. Ing. Gustavo Bastian que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará “Desarrollo de aplicación Hospitalaria con MQTT”. Consistirá esencialmente en la implementación de un prototipo de una aplicación móvil para la llamada, gestión de enfermeras y consulta enfermera-médico mediante la utilización del protocolo MQTT, y tendrá un presupuesto preliminar estimado de 600 hs de trabajo y \$60000, con fecha de inicio 31 de Octubre de 2021 y fecha de presentación pública 15 de mayo de 2022.

Se adjunta a esta acta la planificación inicial.

Dr. Ariel Lutenberg
Director posgrado FIUBA

Ing. Sergio Starkloff
Surix S.R.L

Mg.Ing. Ericson Joseph Estupiñan Pineda
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

En la actualidad, el avance de la Internet de las Cosas (IOT) y la disminución de costos asociados a la tecnología hacen factible su incorporación a distintos campos de la vida cotidiana. Uno de esos campos es el de infraestructuras hospitalarias inteligentes.

Por otra parte, dentro de las múltiples opciones para realizar la comunicación entre los dispositivos IOT, el protocolo Message Queuing Telemetry Transport (en adelante MQTT) se ha probado como un protocolo confiable y ampliamente utilizado.

Dentro del contexto, en este trabajo se desarrollará una aplicación multiplataforma que utilizará el protocolo MQTT para los distintos participantes de la actividad hospitalaria. El proyecto es una necesidad de la empresa Surix S.R.L. y se lleva a cabo como parte de la carrera Especialización de Internet de las Cosas.

Surix S.R.L. es una firma que se dedica al desarrollo, fabricación y comercialización de productos IP y sistemas hospitalarios de calidad. Posee una comprobada trayectoria dentro del mercado local e internacional. Se destaca por su compromiso con la industria nacional, la mejora continua de sus productos y el soporte que brinda a sus clientes. Este proyecto se enmarca dentro del segundo ítem de su misión, porque mejora y extiende capacidades a un sistema existente.

Surix S.R.L. fabrica un sistema IP de llamado a enfermera que está basado en el protocolo SIP. Este consiste en un servidor central y terminales que se encuentran en las habitaciones del hospital. La aplicación principal se ejecuta en una pc o bien en una tablet y monitorea el estado de las habitaciones.

El objetivo del proyecto es realizar un sistema con las ventajas que provee el protocolo MQTT, como es la posibilidad de agregar accesorios rápidamente con bajo costo de software, hardware e implementación.

MQTT es un protocolo open source liviano, hecho que permite aplicarlo en dispositivos con pocos recursos y baja velocidad de transmisión, ampliamente utilizado en dispositivos IOT. Está basado en la pila TCP/IP, se implementa en la capa de aplicación y sus mensajes se transmiten como colas de publicación/subscripción.

El desafío de este proyecto consiste en la programación de un sistema que contenga un servidor o broker MQTT, una base de datos donde alojar información de reportes de habitaciones/enfermeras y datos relevantes al paciente (incluyendo temporizadores para suministro de ciertos medicamentos y/o control), una página web para configuración y una aplicación multiplataforma donde se realice la gestión de datos e interacción con los clientes. La aplicación será capaz de identificar la cama correspondiente (mediante lectura de símbolos QR) y de transmitir mensajes de voz en caso de ser necesario.

La motivación que origina la realización del proyecto es generar las bases para poder incorporar otros dispositivos inteligentes al sistema principal a bajo costo. Por ejemplo, en un futuro se puede monitorear la temperatura de la habitación y saber si sufre un desperfecto el aire acondicionado, escuchar sonidos dentro de la sala en caso de que el paciente no pueda acceder al llamador, etc.

El proyecto completo consta de :

- Broker.
- Pantalla web de configuración.
- Cliente médico.
- Cliente enfermera.
- Cliente sistema.

En la figura 1 se presenta un diagrama de interconexión entre los dispositivos.

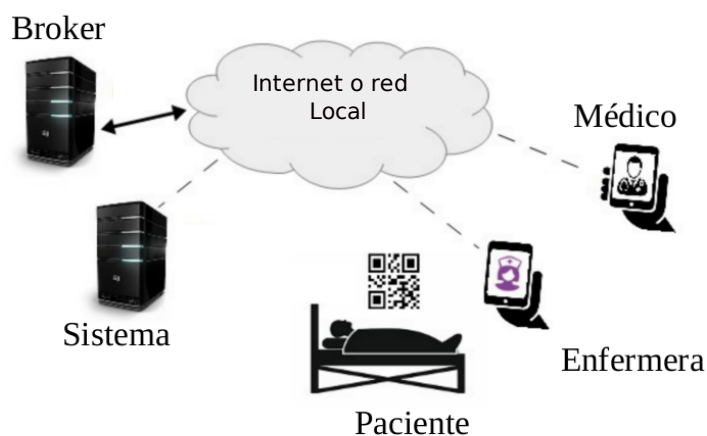


Figura 1. Diagrama en bloques del sistema.

El broker recibe mensajes llamados “eventos”, de distintos publicadores y los reenvía a los subscriptores que correspondan según una política asignada previamente. El cliente sistema puede gestionar la base de datos con información relevante para los pacientes y también puede observar el estado de las habitaciones. Mientras que el cliente enfermera solo recibe asignaciones o respuestas de un médico, y publica finalización o consulta (por medio de la aplicación). Por último, el cliente médico solo recibe consultas y publica respuestas por medio de la aplicación.

Con la aplicación web se puede cargar la base de datos con información de enfermeras, pacientes y médicos.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Ing. Sergio Starkloff	Surix S.R.L	Propietario
Responsable	Ing. Gustavo Bastian	FIUBA	Alumno
Colaboradores	–	–	–
Orientador	Mg.Ing. Ericson Joseph Estupiñan Pineda	Surix S.R.L	Director trabajo final
Usuario final	Hospitales, personal de salud, administradores de sistemas	–	–

- Orientador: Ericson va a poder ayudar mucho con la definición de los requerimientos.
- Usuario final: Todos los usuarios del sistema como ser, administradores de redes hospitalarias, médicos y enfermeras, que deseen observar y/o controlar el proceso.

3. Propósito del proyecto

El propósito de este proyecto es desarrollar un sistema para gestión de enfermeras basado en el protocolo MQTT compuesto por un broker, una web de configuración y una o varias aplicaciones, que agilice el desarrollo de funcionalidades futuras.

4. Alcance del proyecto

El presente proyecto incluye:

- Confección del plan de trabajo.
- Investigación y estudio del protocolo MQTT, bases de datos SQL y NoSQL, programación de aplicaciones web y programación de aplicaciones multiplataforma.
- Desarrollo local del broker que registre en un log las actividades realizadas y su horario de realización.
- Desarrollo local de una página web de configuración.
- Desarrollo local de una aplicación cliente médico con envío/recepción de mensajes de audio/texto/alarmas.
- Desarrollo local de una aplicación cliente enfermera con envío/recepción de mensajes de audio/texto/alarmas y escaneo de QR para identificar paciente.
- Desarrollo local de una aplicación que asigna enfermeras a pacientes de acuerdo a la indicación de un médico. Por ejemplo, todos los días a las 17 hs, medir la presión arterial del paciente “x”, independientemente de la enfermera. Esta aplicación permite monitorizar el estado de las habitaciones.
- Documentación de las aplicaciones desarrolladas.

El presente proyecto NO incluye:

- Manuales de las distintas aplicaciones desarrolladas.
- Traducciones a idiomas extranjeros de las aplicaciones y/o de la página web.
- Sistema llamador del paciente.
- Análisis en profundidad de tráfico en la red.
- Análisis en profundidad de Seguridad.
- Contratación de base de datos remota.
- Contratación e instalación de servidores remotos.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- Existe un llamador paciente.
- Las aplicaciones moviles usaran iOS y Android como sistema operativo.
- La base de datos de los pacientes sólo puede ser afectada por medio de la aplicación sistema (el cliente médico sólo puede hacer consultas/modificaciones puntuales a una situación).
- Se dispondrá al menos de una pc para instalación del servidor web, la base de datos, y del broker MQTT.
- Se dispondrá de un router para la prueba funcional del sistema.

6. Requerimientos

Los principales requerimientos relevados son los siguientes:

1. Requerimientos funcionales del Servidor:

- 1.1. El servidor debe tener instalado Eclipse Mosquitto como broker MQTT.
- 1.2. El servidor debe tener endpoints para interactuar con una aplicación Web, para consultas y envío de datos hacia dispositivos clientes.
- 1.3. El servidor debe tener endpoints para interactuar con la base de datos.

2. Requerimientos funcionales de la Base de datos:

- 2.1. El sistema debe poseer una base de datos relacional.
- 2.2. La base de datos debe poseer datos cargados por default.
- 2.3. La base de datos a utilizar debe poseer las siguientes tablas:
 - Habitaciones
 - Pacientes

- Médicos
- Enfermeras
- Eventos

3. Requerimientos funcionales de la aplicación web de configuración:

- 3.1. La aplicación web debe ser cliente del broker MQTT.
- 3.2. La aplicación web debe poseer funciones de consulta o modificación de la base de datos.
- 3.3. La aplicación web debe permitir visualización de estadísticas de pacientes o enfermeras.
- 3.4. La aplicación debe contener acceso con usuario y contraseña para cada persona.

4. Requerimientos funcionales de la aplicación enfermera:

- 4.1. La aplicación enfermera debe tener 3 modos de uso: usuario médico, usuario enfermera y usuario sistema.
- 4.2. Al logearse el usuario , y contrastarse con la base de datos del sistema, se activa el modo correspondiente.
- 4.3. La aplicación en modo enfermera debe permitir leer códigos QR.
- 4.4. La aplicación en modo enfermera debe poder descargar información relevante del paciente (medicamentos a suministrar, controles a realizar, contacto del médico a cargo del tratamiento, etc).
- 4.5. La aplicación en modo sistema debe mostrar las habitaciones sin atención, según una tabla de prioridades y en caso de igualdad de prioridades, mostrár según el orden de llamada.
- 4.6. El modo usuario médico y el modo usuario enfermera deben poder enviar mensajes de textos o sonido comprimido en 32kbps (mp3).

5. Requerimientos de documentación:

- 5.1. Confección de documento con información relativa a la base de datos: detalles de la misma y de API para acceder.
- 5.2. Diagramas UML de aplicación sistema, aplicación enfermera, aplicación médico y página web.

6. Requerimientos de integración de sistema:

- 6.1. El sistema debe integrar el funcionamiento del servidor con base de datos , aplicación web y aplicaciones móviles.

7. Historias de usuarios (*Product backlog*)

Se identifican los siguientes roles:

- Usuario final: persona que utilizará el sistema. Existen 4 usuarios finales bien definidos: usuario final modo enfermero, usuario final modo médico y usuario final modo operador del sistema.
- cliente: persona que necesita el desarrollo, sus intereses están orientados al negocio.

Story points: Se utilizarán 1, 2, 3, 5 y 8 para la estimación de las historias de usuario.

- Historias de 1 punto: son historias cuya implementación es rápida y con baja posibilidad de falla. Por ejemplo, envío de un mensaje sencillo entre un nodo y el servidor de la red.
- Historias de 2 puntos: son historias cuya implementación no es rápida y pero tienen baja posibilidad de falla. Por ejemplo, diseño de la interfaz de usuario.
- Historias de 3 puntos: son historias cuya implementación no es rápida y posibilidad de falla media. Por ejemplo, lectura de códigos QR, hay que investigar distintas librerías y ver el correcto funcionamiento en distintos dispositivos.
- Historias de 5 puntos: son historias cuya implementación es lenta y con posibilidad de falla alta ya que interactúan varios elementos de la red. Por ejemplo, envío de un mensaje de audio entre distintos actores, debido a que hay que corroborar el funcionamiento de la captura de audio en ambos dispositivos, el correcto traslado por la red y la recepción perfecta del archivo.
- Historias de 8 puntos: son historias cuya implementación es compleja y con mucha posibilidad de falla ya que interactúan distintos componentes del sistema. Por ejemplo, solicitud de ayuda de una enfermera al sistema, debe chequear disponibilidad de otras enfermeras, y dependiendo del tipo de ayuda necesaria, las capacidades de la misma.

Historias de Usuario:

- Como usuario en modo enfermero quiero que la aplicación me notifique si hay un pedido de una habitación para ir a atenderla. (1 punto)
- Como usuario en modo enfermero quiero que la aplicación permita que notifique que finalicé la tarea para poder recibir otro pedido. (1 punto)
- Como usuario en modo enfermero quiero que la aplicación me permita notificar que estoy atendiendo para que no tener varios pedidos pendiente. (3 puntos)
- Como usuario en modo enfermero quiero que la aplicación permita que consulte a un médico en caso de necesidad. (8 puntos)
- Como usuario en modo enfermero quiero que la aplicación permita obtener información del paciente para entender el estado de la situación. (5 puntos)
- Como usuario en modo enfermero quiero que la aplicación permita avisar que mi jornada terminó para no recibir más notificaciones. (5 puntos)
- Como usuario en modo enfermero quiero que la aplicación permita que notifique que necesito ayuda de otro enfermero para atender el paciente. (8 puntos)
- Como usuario en modo médico quiero que la aplicación posea un mecanismo sencillo para responder dudas de la enfermera. (3 puntos)
- Como usuario en modo médico quiero que la aplicación permita acceder a la información del paciente que en caso de consulta de un enfermero. (2 puntos)
- Como usuario en modo sistema quiero que la aplicación me permita asignar tareas temporizadas para atender rutinas preestablecidas por el médico. (5 puntos)

- Como usuario en modo sistema quiero que la aplicación permita obtener datos estadísticos de cada enfermero para poder reportar. (2 puntos)
- Como usuario en modo sistema quiero que la aplicación móvil me muestre el estado de las habitaciones. (1 punto)
- Como cliente quiero que el sistema sea de fácil instalación. (1 punto)

8. Entregables principales del proyecto

Los entregables del proyecto son:

- Plan de proyecto del Trabajo Final y Memoria Técnica.
- Servidor MQTT.
- Base de datos configurada.
- Aplicación Web responsive.
- Aplicación enfermera (con los 3 modos de funcionamiento).
- Código del sistema(utilizando dockerhub y github).

9. Desglose del trabajo en tareas

Desglose del trabajo:

1. Investigación preliminar (60 hs).
 - 1.1. Investigar funcionamiento del Broker Mosquitto (10 hs).
 - 1.2. Investigar base de datos relacionales (10 hs).
 - 1.3. Investigar bibliotecas gráficas para implementar página web (20 hs).
 - 1.4. Investigar bibliotecas gráficas para implementar aplicación móvil (20 hs).
2. Implementación del servidor web(40 hs).
 - 2.1. Instalación y configuración de Node.js(10 hs).
 - 2.2. Instalación del broker MQTT (10 hs).
 - 2.3. Programación de funciones de consulta para aplicación web /móvil (20 hs).
3. Implementación de la base de datos (110 horas).
 - 3.1. Instalación de base de datos (40 horas).
 - 3.2. Configuración de parámetros de seguridad y datos (30 horas).
 - 3.3. Creación y carga de datos en tablas(40 horas).
4. Implementación de página web (190 horas).
 - 4.1. Programación de funciones MQTT como cliente del broker (50 hs).

- 4.2. Prueba de envío y recepción de datos al broker (10 hs).
- 4.3. Programación de funciones de consulta de datos a la base de datos (CRUD parte 1) (40 hs).
- 4.4. Programación de funciones de carga de datos a la base de datos (CRUD parte 2) (40 hs).
- 4.5. Programación de lógica de acceso y seguridad de usuarios (40 hs).
- 4.6. Prueba de acceso de usuarios (10 hs).
- 5. Implementación de aplicación enfermera (140 horas).
 - 5.1. Programación de interfaz de login de usuarios (8 hs).
 - 5.2. Testeo de lógica y acceso de seguridad de usuarios (6 hs).
 - 5.3. Programación de interfaces de usuario aplicación móvil (30 hs).
 - 5.4. Programación de funciones de acceso a la base de datos (40 hs).
 - 5.5. Prueba de envío y recepción de datos entre clientes modo enfermera y cliente modo médico (10 hs).
 - 5.6. Programación y prueba de lectura de códigos QR (6 hs).
 - 5.7. Programación de captura/compresión/envío de audio usuario modo enfermera (30 hs).
 - 5.8. Prueba de envíos de distintos audios (10 hs).
- 6. Pruebas de sistema (10 hs).
 - 6.1. Prueba del funcionamiento correcto con 2 enfermeras 2 médicos y 3 llamadores (10 hs).
- 7. Memoria técnica del proyecto (50 hs).
 - 7.1. Redacción de la memoria técnica del proyecto(50 hs).

Cantidad total de horas: (600 hs).

10. Diagrama de Activity On Node

En la siguiente figuras se identifican mediante un color las distintas etapas del proyecto:

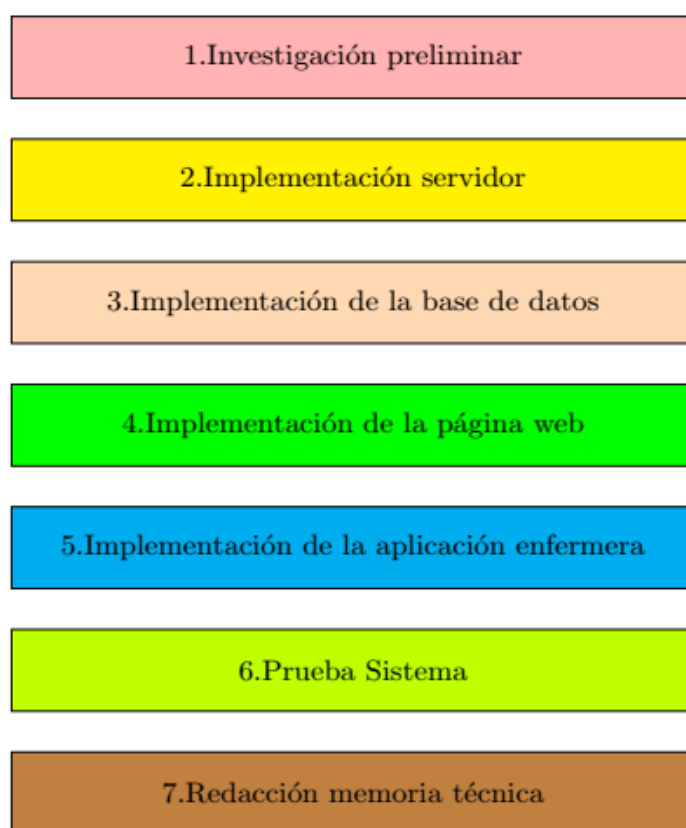


Figura 2. Colores en *Activity on Node*

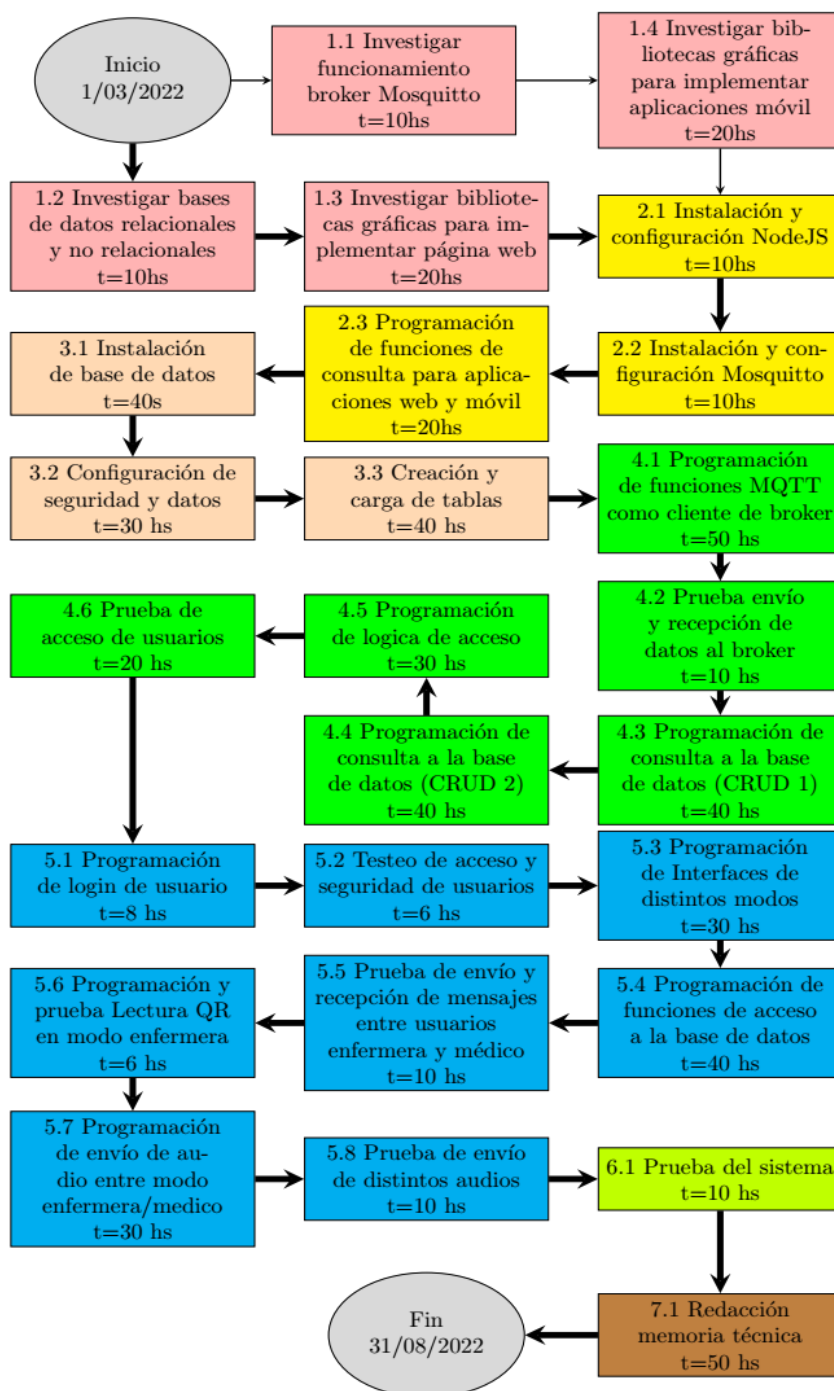


Figura 3. Diagrama en *Activity on Node*

11. Diagrama de Gantt

En la siguiente figura se presenta la lista de tareas ingresadas al sistema de gestion de tareas, un diagrama de gantt reducido, con las tareas acordes con el diagrama Activity on Node y otro con todas las tareas.

Nombre	Inicio	Fin
<u>1 Investigación preliminar.</u>	1/3/22	3/3/22
1.1 Investigar funcionamiento broker Mosquitto.	1/3/22	1/3/22
1.2 Investigar bases de datos relacionales y no relacionales	1/3/22	1/3/22
1.3 Investigar bibliotecas para desarrollo página web.	2/3/22	3/3/22
1.4 Investigar bibliotecas para aplicación móvil.	2/3/22	3/3/22
Herramientas definidas	4/3/22	4/3/22
<u>2 Implementar servidor.</u>	4/3/22	8/3/22
2.1 Instalación y configuración Node.Js.	4/3/22	4/3/22
2.2 Instalación broker MQTT.	5/3/22	5/3/22
2.3 Programación de funciones de consulta para aplicación web/móvil.	7/3/22	8/3/22
<u>3 Implementar base de datos.</u>	9/3/22	12/4/22
3.1 Instalación base de datos.	9/3/22	19/3/22
3.2 Configuración de parámetros de seguridad y datos.	21/3/22	30/3/22
3.3 Creación y carga de datos en tabla.	31/3/22	12/4/22
Sistema listo para desarrollar	13/4/22	13/4/22
<u>4 Implementación de página web.</u>	13/4/22	3/6/22
4.1 Programación de funciones MQTT como cliente del broker.	13/4/22	23/4/22
4.2 Prueba de envío y recepción de datos al broker.	25/4/22	26/4/22
4.3 Programación de consultas a base de dato(CRUD 1).	27/4/22	7/5/22
4.4 Programación de carga de datos en la base de datos (CRUD 2).	9/5/22	19/5/22
4.5 Programación de logica de acceso y seguridad de usuarios.	20/5/22	1/6/22
4.6 Prueba de acceso de usuarios.	2/6/22	3/6/22
Finalización pagina web configuración	4/6/22	4/6/22
<u>5 Implementación de aplicación enfermera.</u>	4/6/22	14/7/22
5.1 Programación de interfaz de login de usuarios.	4/6/22	6/6/22
5.2 Testeo de lógica y acceso de seguridad de usuarios.	7/6/22	7/6/22
5.3 Programación de interfaces de usuario de aplicación móvil.	8/6/22	16/6/22
5.4 Programación de funciones de acceso a la base de datos.	17/6/22	28/6/22
5.6 Programación y prueba de lectura de códigos QR.	29/6/22	29/6/22
5.5 Prueba de envío y recepción de datos entre clientes modo enfermera y cliente modo médico.	30/6/22	1/7/22
5.7 Programación de captura/compresión/envío de audio modo enfermera.	2/7/22	12/7/22
5.8 Prueba de envío de distintos audios.	13/7/22	14/7/22
Finalización primer prototipo aplicación enfermera	15/7/22	15/7/22
<u>6 Pruebas de sistema.</u>	15/7/22	16/7/22
6.1 Prueba de funcionamiento correcto con 2 enfermera 2 médico y 3 llamadores.	15/7/22	16/7/22
<u>7 Memoria técnica del proyecto</u>	18/7/22	28/7/22
Redacción de la memoria técnica del proyecto	18/7/22	28/7/22

Figura 4. Nombre de tareas

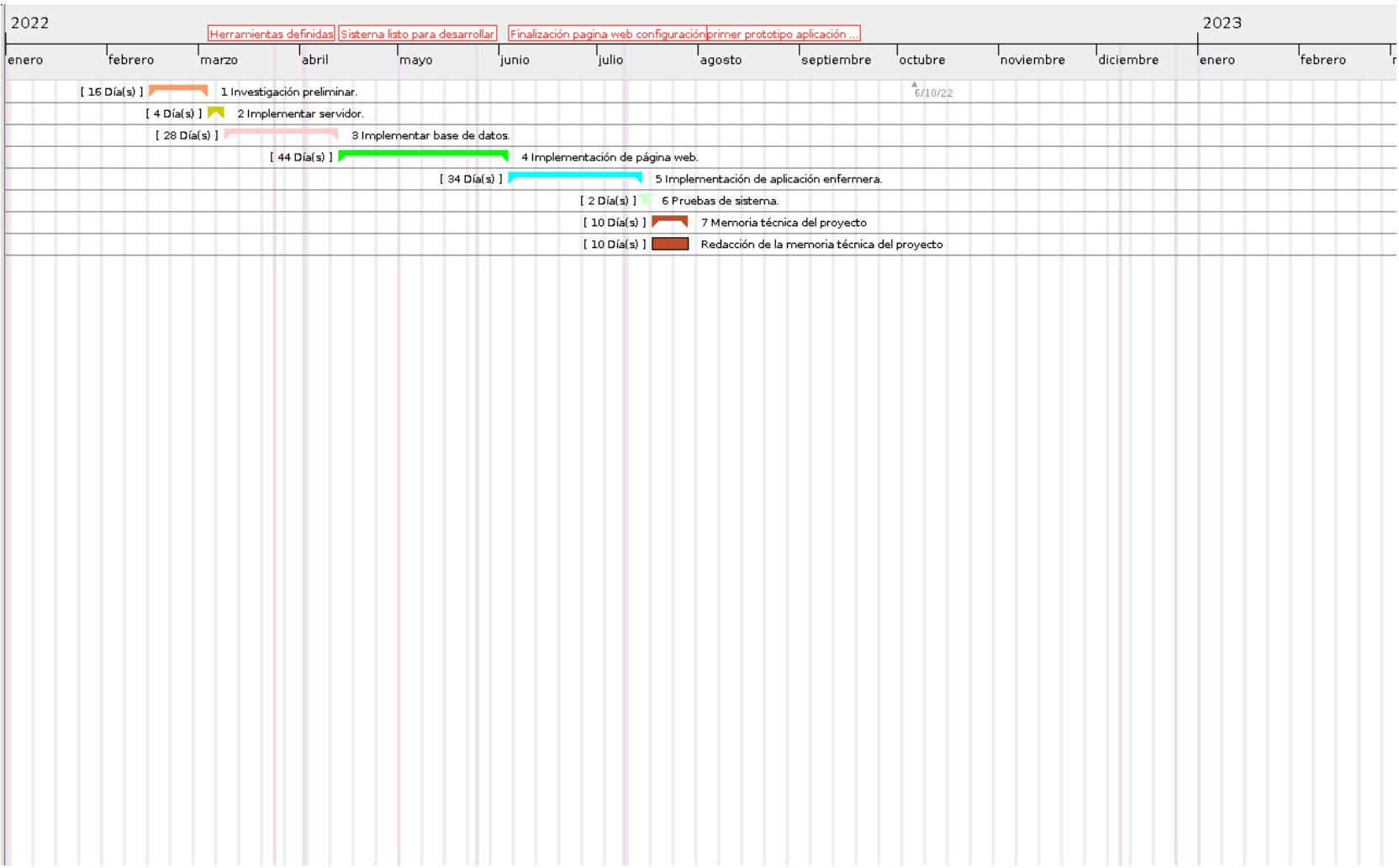


Figura 5. Diagrama de Gantt Tareas principales

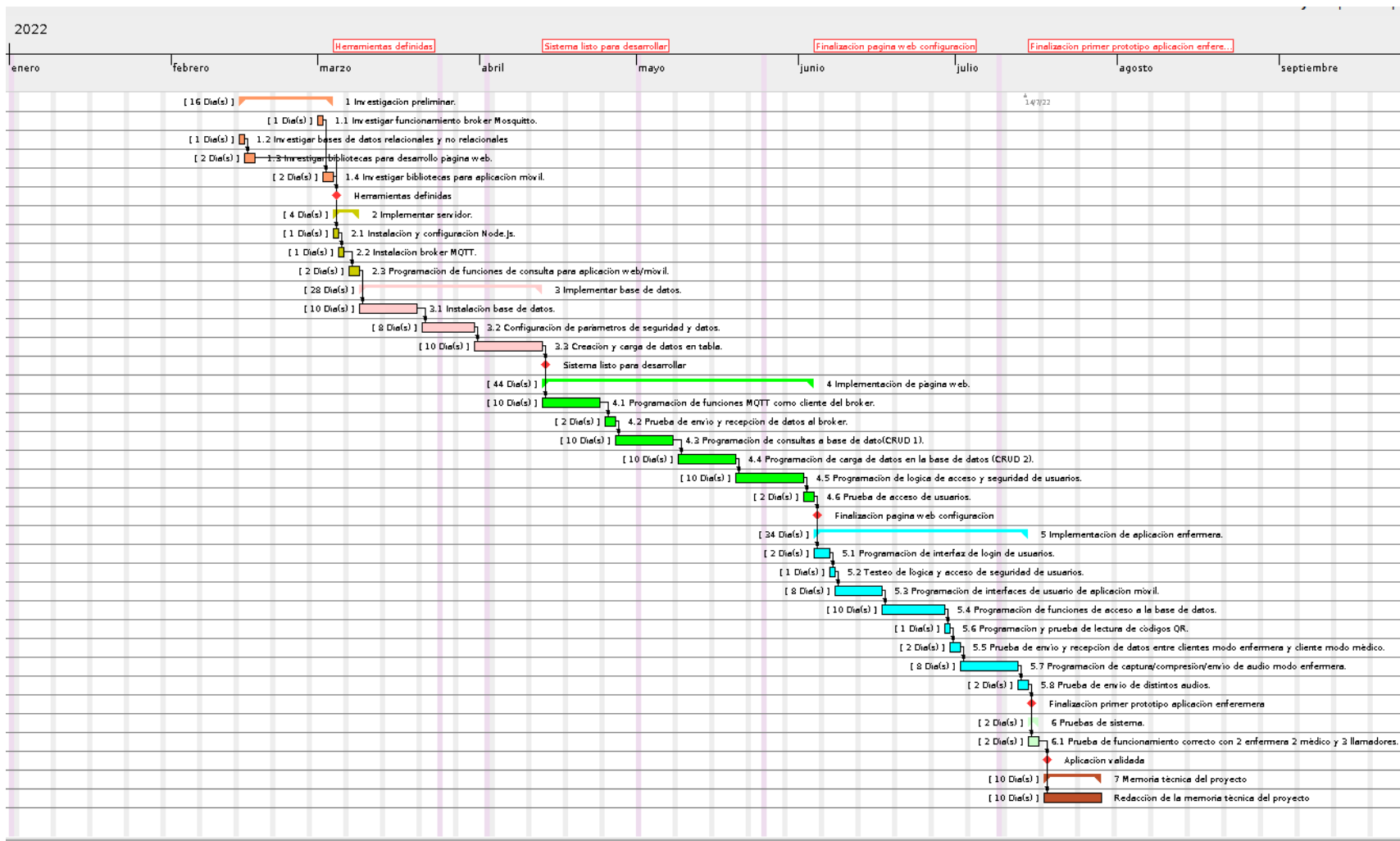


Figura 6. Diagrama de Gantt

12. Presupuesto detallado del proyecto

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Investigación	60hs	1000	60000
Programación	424hs	1000	424000
Testing	66hs	1000	66000
Documentación	50hs	1000	50000
SUBTOTAL			600000
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Raspberry pi 4(Server)	1	27000	27000
Costos de Luz eléctrica	600	4	2400
Asesorías	20hs	1000	20000
SUBTOTAL			49400
TOTAL			649400

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 $RPN=S \times O$)

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

Riesgo	S	O	RPN	S*	O*	RPN*

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.