



Gestión remota de calefactor eléctrico para el hogar

Autor:

Ing. Leonardo Mancini

Director:

Mg. Ing. Diego Javier Brengi (INTI)

Codirector:

Ing. Salvador Tropea (INTI)

Esta planificación fue realizada en el curso de Gestión de proyectos entre el 26 de abril de 2022 y el 14 de junio de 2022.

Índice

| | |
|--|-----------|
| 1. Descripción técnica-conceptual del proyecto a realizar | 5 |
| 1.1 Detalle de funcionalidad a implementar | 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 | 7 |
| 7. Historias de usuarios (<i>Product backlog</i>). | 10 |
| 8. Entregables principales del proyecto | 12 |
| 9. Desglose del trabajo en tareas | 13 |
| 10. Diagrama de Activity On Node. | 15 |
| 11. Diagrama de Gantt | 16 |
| 12. Presupuesto detallado del proyecto | 21 |
| 13. Gestión de riesgos | 22 |
| 14. Gestión de la calidad | 24 |
| 15. Procesos de cierre | 29 |

Registros de cambios

| Revisión | Detalles de los cambios realizados | Fecha |
|----------|---|---------------------|
| 0 | Creación del documento | 26 de abril de 2022 |
| 1 | Se completa hasta el punto 5 inclusive | 10 de mayo de 2022 |
| 2 | Se completa hasta el punto 9 inclusive | 17 de mayo de 2022 |
| 3 | Se completa hasta el punto 12 inclusive | 24 de mayo de 2022 |
| 4 | Se completa hasta el punto 15 inclusive | 31 de mayo de 2022 |

Acta de constitución del proyecto

Buenos Aires, 26 de abril de 2022

Por medio de la presente se acuerda con el Ing. Leonardo Mancini que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará “Gestión remota de calefactor eléctrico para el hogar”, consistirá esencialmente en la implementación de un prototipo de un sistema de conexión a internet de calefactor eléctrico, y tendrá un presupuesto preliminar estimado de 600 hs de trabajo y AR\$1251500, con fecha de inicio 26 de abril de 2022 y fecha de presentación pública 16 de abril de 2023.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg
Director posgrado FIUBA

Sr. Pablo Barbero
Intelligentgas

Mg. Ing. Diego Javier Brengi
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

El presente proyecto plantea desarrollar una solución IoT (Internet of Things) para calefactores eléctricos comercializados por la empresa Intelligentgas. El proyecto es realizado como parte del trabajo final de la Especialización en Internet de las Cosas mediante convenio con el INTI (Instituto Nacional de Tecnología Industrial). Actualmente la empresa comercializa soluciones IoT enfocadas en el control y monitoreo de sensores de diversos gases. La empresa tiene como objetivo ingresar al mercado de los smart home desarrollando electrodomésticos con características de IoT. El producto que se encuentra desarrollando en este momento es un calefactor eléctrico. En el mercado argentino no existen productos similares por lo que para la empresa es un nicho en el cual innovar. La posibilidad de configurar en forma remota, monitorear su uso y consumo lo dotarán de una característica importante, tanto por una cuestión ambiental como comercial. Para lograr este objetivo es que se da inicio al proyecto de brindarle conectividad a Internet mediante el desarrollo del módulo de comunicaciones. Cabe destacar que la comunicación no es el único objetivo de la solución ya que un valor agregado directamente para el cliente es la de poder tener datos que le permitirán conocer las costumbres de uso de los usuarios. Por lo tanto la solución planteada es una infraestructura IoT además del desarrollo del módulo Wi-Fi.

1.1. Detalle de funcionalidad a implementar

La solución planteada consta de:

- Módulo de comunicación Wi-Fi que permite conectar el calefactor a Internet. Mediante este módulo el calefactor podrá ser configurado y enviar datos al usuario final. La comunicación será mediante el protocolo MQTT cuyo broker se encontrará en un servidor web.
- Servidor en la nube en donde se encontrará la base de datos con información de cada calefactor y de cada usuario.
- Aplicación móvil mediante la cual el usuario podrá interactuar con los calefactores de su propiedad.

El desarrollo del sistema permitiría a cada usuario configurar y monitorear los calefactores de su propiedad. También permitirá encender/apagar los calefactores en determinado rango de tiempo y/o por distancia geográfica (si deja el hogar/está volviendo al hogar). En la figura 1 se presenta el diagrama en bloques del sistema, sus principales componentes y como interactúan.

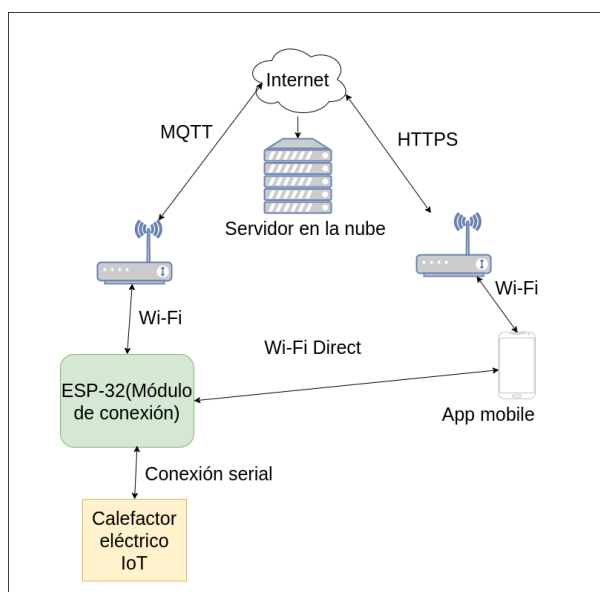


Figura 1. Diagrama en bloques del sistema.

2. Identificación y análisis de los interesados

| Rol | Nombre y Apellido | Organi-zación | Puesto |
|---------------|------------------------------|----------------|------------------------|
| Cliente | Sr. Pablo Barbero | Intelligentgas | - |
| Responsable | Ing. Leonardo Mancini | FIUBA | Alumno |
| Orientador | Mg. Ing. Diego Javier Brengi | INTI | Director Trabajo final |
| Usuario final | Usuarios de calefactores IoT | - | - |

- Director: muy detallista en los procesos de gestión de proyectos. Se programan las reuniones mediante meet durante el mediodía en forma semanal.
- Cliente: tiene conocimientos técnicos, ya cuenta con desarrollos similares. Se puede comunicar por teléfono en cualquier momento. También solicita guardar confidencialidad en detalles técnicos/comerciales al ser un producto comercial.

3. Propósito del proyecto

El propósito de este proyecto es brindar funcionalidad IoT a los calefactores comercializados por la empresa del cliente. El objetivo se logrará con el desarrollo de un módulo de comunicaciones Wi-Fi que permitirá enviar y recibir eventos entre un servidor en la nube y el calefactor. De esta forma se podrá monitorear e interactuar en forma remota y lograr un uso más eficiente del calefactor.

4. Alcance del proyecto

El desarrollo del presente proyecto incluye:

- Desarrollo de firmware para el módulo de conexión Wi-Fi.
- Desarrollo de una aplicación móvil híbrida para el usuario final que permitirá monitorear/configurar el calefactor.
- Desarrollo del servidor en la nube.
- Construcción del prototipo de hardware.

El desarrollo no incluirá:

- Construcción del producto final.
- Gestión de usuarios con Third Party.
- Distribución comercial de las aplicaciones.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- El cliente proveerá el hardware controlador del calefactor.
- El cliente proveerá el hardware de conexión Wi-Fi (Módulo ESP-32).
- El cliente proveerá acceso a los servidores de Google Cloud Platform.
- Durante el cursado de la especialidad se obtendrán los conocimientos necesarios para lograr el objetivo.
- Se dispondrá del acceso a los repositorios de Android/iOS para la publicación.

6. Requerimientos

1. Requerimientos funcionales:

- 1.1. El usuario podrá dar de alta, baja y modificar información (dirección, nombre y descripción) de los calefactores en el sistema.
- 1.2. El calefactor deberá medir la temperatura ambiente en grados centígrados y con una precisión de $\pm 0.5^\circ \text{C}$.
- 1.3. El usuario podrá establecer el valor de la temperatura del calefactor en grados centígrados.
- 1.4. El usuario podrá configurar hora y fecha de encendido y apagado de cada calefactor.
- 1.5. El usuario podrá configurar una distancia de apagado/encendido automático en un rango de 1-50 km.

- 1.6. El usuario podrá establecer el valor de kWh (Kilo-watt hora).
- 1.7. El usuario podrá consultar el consumo de energía en kWh en el mes en curso.
- 1.8. Se deberá registrar las coordenadas geográficas de cada calefactor en el sistema.
2. Requerimientos del módulo embebido:
 - 2.1. Se podrá configurar la red Wi-Fi.
 - 2.2. Se podrá configurar el reloj interno.
 - 2.3. La comunicación con la electrónica del calefactor deberá ser serial.
 - 2.4. Deberá informar los datos de temperatura al servidor cada 10 segundos.
 - 2.5. En caso de mal funcionamiento, no detectar temperatura, informará el error mediante MQTT al servidor.
3. Requerimientos de comunicaciones:
 - 3.1. El calefactor permitirá ser configurado mediante Wi-Fi Direct.
 - 3.2. El calefactor podrá ser accedido mediante Internet para ser configurado y mostrar la temperatura actual.
 - 3.3. El protocolo de comunicación entre servidor y calefactor deberá ser bi-direccional mediante MQTT.
 - 3.4. Las rutas MQTT para identificar cada calefactor deberán seguir una jerarquía: /usuario/usuario/ubicacion/ubicacion/calefactor/calefactor.
 - 3.5. La comunicación entre el dispositivo móvil y el servidor deberá ser mediante APIs REST.
 - 3.6. El servidor, si está configurado el aviso por cercanía, enviará el evento al calefactor.
4. Requerimientos de la aplicación móvil:
 - 4.1. La aplicación móvil deberá mostrar un listado de calefactores del usuario conectado.
 - 4.2. La aplicación móvil deberá mostrar la temperatura actual del ambiente y los valores de configuración del calefactor.
 - 4.3. La aplicación móvil permitirá ingresar el costo del kWh en el sistema para calcular costos de consumo.
 - 4.4. La aplicación móvil mostrará el consumo en kW y el costo en pesos acumulados en el mes actual.
 - 4.5. La aplicación móvil permitirá recibir y mostrar notificaciones cuando ocurre un evento con alguno de los calefactores.
 - 4.6. Deberá tener una pantalla de ingreso a la aplicación.
 - 4.7. La aplicación móvil deberá capturar las coordenadas de GPS con una frecuencia de 10 minutos.
 - 4.8. La aplicación móvil deberá calcular si se encuentra geográficamente en el rango de la ubicación del calefactor y en caso afirmativo enviar evento al servidor.
5. Requerimientos no funcionales:
 - 5.1. El módulo de comunicaciones deberá permitir la configuración de red Wi-Fi mediante WPS.
 - 5.2. El servidor, con el broker y su base de datos deberán estar alojados en un entorno cloud.

- 5.3. Los datos enviados por MQTT deberán estar cifrados con TLS.
- 5.4. La comunicación con la API Rest deberá ser mediante HTTPS.
- 5.5. La aplicación móvil deberá utilizar componentes de "Material Design".

7. Historias de usuarios (*Product backlog*)

La ponderación de las historias de usuario se realizará a través de story points. Para determinar el puntaje para cada historia se evaluará cada una de ellas de acuerdo a tres aspectos: cantidad de trabajo a realizar, complejidad e incertidumbre. Por cada aspecto se utilizarán las siguientes escalas:

- Cantidad de trabajo:

- Alta: 8
- Media: 5
- Baja 1

- Complejidad:

- Alta: 8
- Media: 5
- Baja: 1

- Incertidumbre:

- Alta: 5
- Media: 3
- Baja: 0

Se utilizará la escala de Fibonacci para determinar el puntaje final: 0,1,1,2,3,5,8,13,21. Utilizando estas escalas y evaluando los tres aspectos, se procede a sumar el puntaje final, redondeando al próximo valor de Fibonacci más cercano. Si el puntaje supera los 21 puntos, la historia deberá dividirse en historias más pequeñas.

1. Como usuario del calefactor quiero configurar la temperatura del mismo para que el calefactor pueda calefaccionar el ambiente donde se ubica.
 - Cantidad de trabajo: media
 - Complejidad: media
 - Incertidumbre: baja
 - Puntos: 13
2. Como usuario del calefactor quiero configurar hora de encendido y apagado para poder programar la calefacción del hogar.
 - Cantidad de trabajo: media
 - Complejidad: media
 - Incertidumbre: media
 - Puntos: 13
3. Como usuario del calefactor quiero recibir notificaciones de mal funcionamiento para poder saber que el calefactor ha dejado de funcionar en el momento que suceda.
 - Cantidad de trabajo: alta

- Complejidad: media
 - Incertidumbre: media
 - Puntos: 21
4. Como usuario del calefactor quiero configurar el valor, en pesos, del kWh para que se puedan realizar los gastos en un periodo de tiempo.
- Cantidad de trabajo: baja
 - Complejidad: baja
 - Incertidumbre: baja
 - Puntos: 2
5. Como usuario del calefactor quiero obtener el consumo de electricidad mensual para poder obtener los gastos en un periodo de tiempo.
- Cantidad de trabajo: media
 - Complejidad: baja
 - Incertidumbre: baja
 - Puntos: 8
6. Como usuario del calefactor quiero activar el apagado automático por distancia geográfica para poder automatizar cuando salgo del hogar.
- Cantidad de trabajo: baja
 - Complejidad: baja
 - Incertidumbre: media
 - Puntos: 5

8. Entregables principales del proyecto

En esta sección se listan los entregables del proyecto. Cada uno de estos items constituye un hito en el proyecto.

- Broker MQTT instalado y configurado.
- Repositorios de código fuente del módulo de comunicaciones, backend y aplicación híbrida.
- Base de datos del servidor.
- API RESTs para el backend.
- Prototipo funcional de la aplicación para el usuario final.
- Memoria final del proyecto.
- Informe de avance.
- Scripts para generación de la aplicación.
- Diagrama de la arquitectura de la solución.

9. Desglose del trabajo en tareas

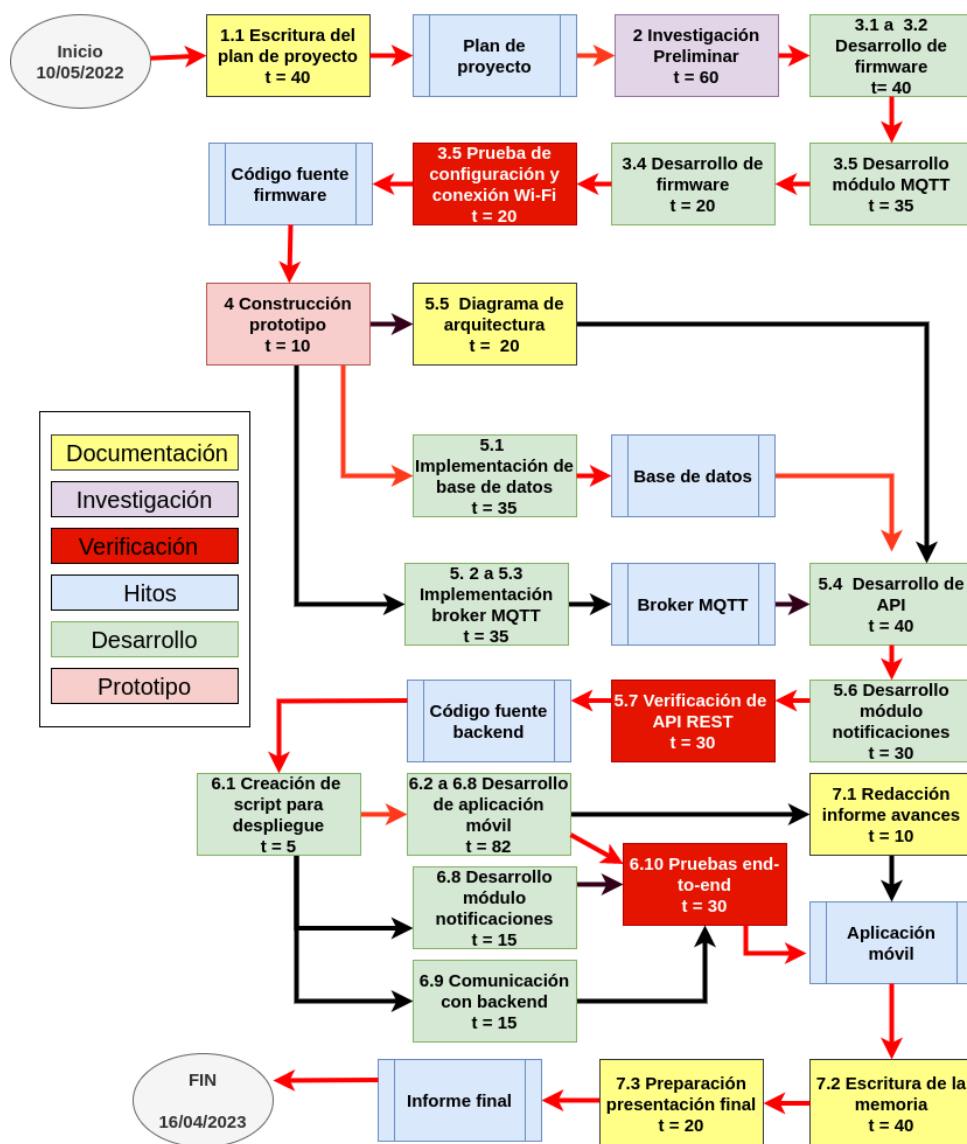
1. Planificación del proyecto.
 - 1.1. Escritura del plan de proyecto (40 hs).
2. Investigación preliminar.
 - 2.1. Investigación de la placa controladora: como conectarla en serie al módulo Wi-Fi y el formato de la trama de datos enviados (30 hs).
 - 2.2. Investigación sobre la configuración WPS del módulo Wi-Fi (20 hs).
 - 2.3. Investigación sobre la configuración del protocolo MQTT en modulo Wi-Fi (10 hs).
3. Desarrollo de firmware (embebido en módulo Wi-Fi).
 - 3.1. Desarrollar módulos de conectividad Wi-Fi y Wi-Fi Direct (20 hs).
 - 3.2. Desarrollar el módulo de comunicación serial con calefactor eléctrico (20 hs).
 - 3.3. Desarrollo de módulo MQTT (35 hs).
 - 3.4. Desarrollo de la configuración del clock interno de módulo Wi-Fi (20 hs).
 - 3.5. Prueba de configuración y conexión del módulo WI-Fi (20 hs).
4. Construcción de prototipo.
 - 4.1. Realizar conectividad por hardware entre el módulo Wi-Fi y la placa controladora del calefactor(4 hs).
 - 4.2. Verificar conectividad y lectura de datos de la controladora (6 hs).
5. Desarrollo de backend.
 - 5.1. Implementación del modelo de base de datos (35 hs).
 - 5.2. Instalación y configuración del broker MQTT (15 hs).
 - 5.3. Pruebas del broker MQTT, verificar conexión con prototipo. (20 hs)
 - 5.4. Desarrollo de la API REST (40 hs).
 - 5.5. Desarrollo del diagrama de arquitectura de la solución (20 hs).
 - 5.6. Desarrollo del módulo de mensajería para envío de notificaciones (30 hs).
 - 5.7. Verificar funcionamiento de la API REST (20 hs).
6. Desarrollo de aplicación móvil.
 - 6.1. Crear script para el despliegue (5 hs).
 - 6.2. Desarrollar la pantalla de ingreso (5 hs).
 - 6.3. Desarrollar la funcionalidad de gestión de calefactores: alta, baja y modificación (30 hs).
 - 6.4. Desarrollar la pantalla de configuración de conectividad del calefactor (Mediante Wi-Fi direct) (15 hs).
 - 6.5. Desarrollar la funcionalidad de configuración de temperatura ambiente y estado del calefactor (10 hs).

- 6.6. Desarrollar la funcionalidad de programación de encendido/apagado del calefactor (20 hs).
- 6.7. Desarrollo de la pantalla de control de encendido/apagado por distancia geográfica (2 hs).
- 6.8. Desarrollo de módulo de recepción de notificaciones (15 hs).
- 6.9. Implementar la comunicación HTTPS con el backend (15 hs).
- 6.10. Verificar el funcionamiento de la aplicación con pruebas de integración end-to-end (30 hs).
- 7. Documentación del proyecto.
 - 7.1. Redacción del informe de avances (10 hs)
 - 7.2. Escritura de memoria final (40 hs).
 - 7.3. Preparación de la presentación del proyecto (20 hs).

Cantidad total de horas: (641 hs).

10. Diagrama de Activity On Node

En la figura 2 se muestra el diagrama de las actividades del proyecto y su camino crítico. La unidad de tiempo utilizada son horas.



Tareas: t en horas y camino crítico indicado en líneas rojas.

Figura 2. Diagrama en *Activity on Node*

11. Diagrama de Gantt

En esta sección se detalla el diagrama de Gantt del proyecto. A continuación se muestra el listado de las tareas e hitos del proyecto.



| Name | Begin date | End date |
|--|------------|----------|
| ▼ 1. Planificación del proyecto | 5/10/22 | 6/6/22 |
| 1.1. Escritura del plan de proyecto | 5/10/22 | 6/6/22 |
| 1.2 Plan de proyecto | 6/7/22 | 6/7/22 |
| ▼ 2. Investigación preliminar | 6/7/22 | 7/18/22 |
| 2.1. Investigación de la placa controladora | 6/7/22 | 6/27/22 |
| 2.2. Investigación sobre la configuración WPS del módulo Wi-Fi | 6/28/22 | 7/11/22 |
| 2.3. Investigación sobre la configuración del protocolo MQTT en modulo Wi-Fi | 7/12/22 | 7/18/22 |
| ▼ 3. Desarrollo de firmware | 7/19/22 | 10/6/22 |
| 3.1. Desarrollar módulos de conectividad Wi-Fi y Wi-Fi Direct | 7/19/22 | 8/1/22 |
| 3.2. Desarrollar el módulo de comunicación serial con calefactor eléctrico | 8/2/22 | 8/15/22 |
| 3.3. Desarrollo de módulo MQTT | 8/16/22 | 9/8/22 |
| 3.4. Desarrollo de la configuración del clock interno de módulo Wi-Fi | 9/9/22 | 9/22/22 |
| 3.5. Prueba de configuración y conexión del módulo WI-Fi | 9/23/22 | 10/6/22 |
| 3.6 Código fuente firmware | 10/7/22 | 10/7/22 |
| ▼ 4. Construcción de prototipo | 10/7/22 | 10/13/22 |
| 4.1. Realizar conectividad por hardware entre el módulo Wi-Fi y la placa controladora d... | 10/7/22 | 10/10/22 |
| 4.2. Verificar conectividad y lectura de datos de la controladora | 10/11/22 | 10/13/22 |
| ▼ 5. Desarrollo de backend | 10/14/22 | 12/12/22 |
| 5.1. Implementación del modelo de base de datos | 10/14/22 | 10/26/22 |
| 5.2. Instalación y configuración del broker MQTT | 10/27/22 | 11/1/22 |
| 5.3. Pruebas del broker MQTT, verificar conexión con prototipo | 11/2/22 | 11/8/22 |
| 5.4. Desarrollo de la API REST | 11/9/22 | 11/23/22 |
| 5.5. Desarrollo del diagrama de arquitectura de la solución | 10/14/22 | 10/20/22 |

Figura 3. Listado de tareas.



| Name | Begin date | End date |
|--|------------|----------|
| 5.3. Pruebas del broker MQTT, verificar conexión con prototipo | 11/2/22 | 11/8/22 |
| 5.4. Desarrollo de la API REST | 11/9/22 | 11/23/22 |
| 5.5. Desarrollo del diagrama de arquitectura de la solución | 10/14/22 | 10/20/22 |
| 5.6. Desarrollo del módulo de mensajería para envío de notificaciones | 11/24/22 | 12/5/22 |
| 5.7. Verificar funcionamiento de la API REST | 12/6/22 | 12/12/22 |
| 5.8 Base de datos , backend y broker MQTT | 12/13/22 | 12/13/22 |
| ▼ 6. Desarrollo de aplicación móvil | 12/13/22 | 4/18/23 |
| 6.1. Crear script para el despliegue | 12/13/22 | 12/13/22 |
| 6.2. Desarrollar la pantalla de ingreso | 12/14/22 | 12/14/22 |
| 6.3. Desarrollar la funcionalidad de gestión de calefactores: alta, baja y modificación | 12/15/22 | 12/26/22 |
| 6.4. Desarrollar la pantalla de configuración de conectividad del calefactor | 12/27/22 | 12/30/22 |
| 6.5. Desarrollar la funcionalidad de configuración de temperatura ambiente y estado d... | 1/2/23 | 1/4/23 |
| 6.6. Desarrollar la funcionalidad de programación de encendido/apagado del calefactor | 1/5/23 | 1/11/23 |
| 6.7. Desarrollo de la pantalla de control de encendido/apagado por distancia geográfica | 1/12/23 | 1/12/23 |
| 6.8. Desarrollo de módulo de recepción de notificaciones | 1/13/23 | 1/18/23 |
| 6.9. Implementar la comunicación HTTPS con el backend | 1/19/23 | 1/30/23 |
| 6.10. Verificar el funcionamiento de la aplicación con pruebas de integración end-to-end | 1/31/23 | 2/9/23 |
| 6.11 Aplicación móvil | 2/10/23 | 2/10/23 |
| ▼ 7. Documentación del proyecto | 12/14/22 | 4/18/23 |
| 7.1. Redacción del informe de avances | 12/14/22 | 12/20/22 |
| 7.2. Escritura de memoria final | 3/15/23 | 4/11/23 |
| 7.3. Preparación de la presentación del proyecto | 4/12/23 | 4/18/23 |
| 7.4 Memoria final | 4/19/23 | 4/19/23 |

Figura 4. Listado de tareas.

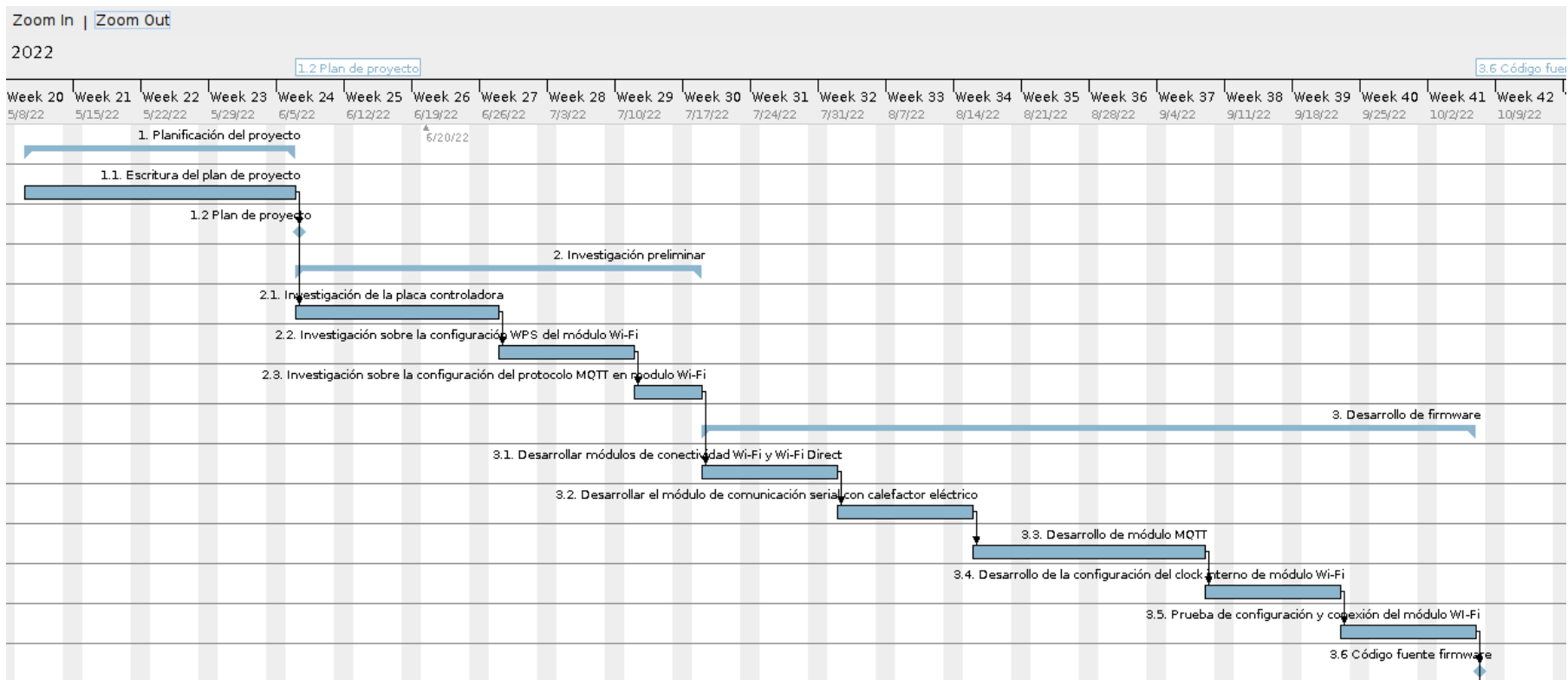


Figura 5. Diagrama de Gantt.

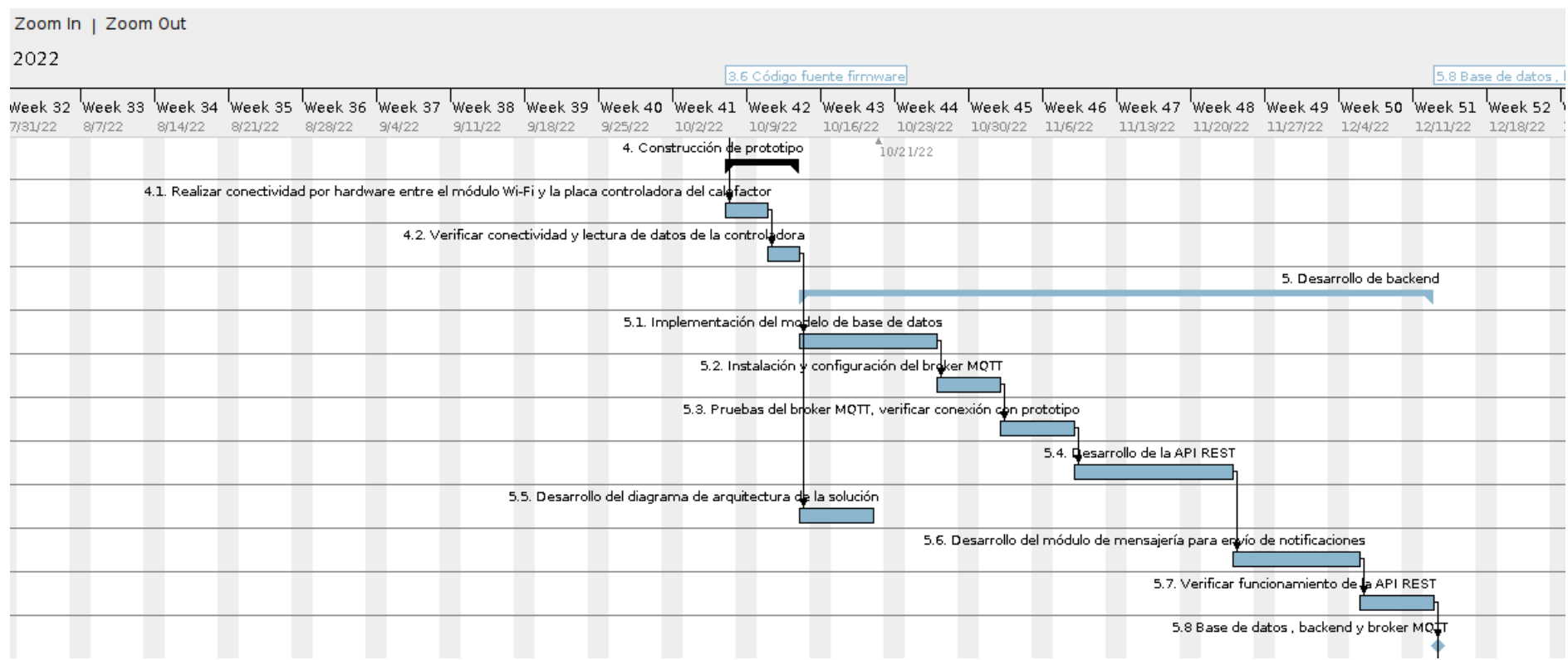


Figura 6. Diagrama de Gantt.

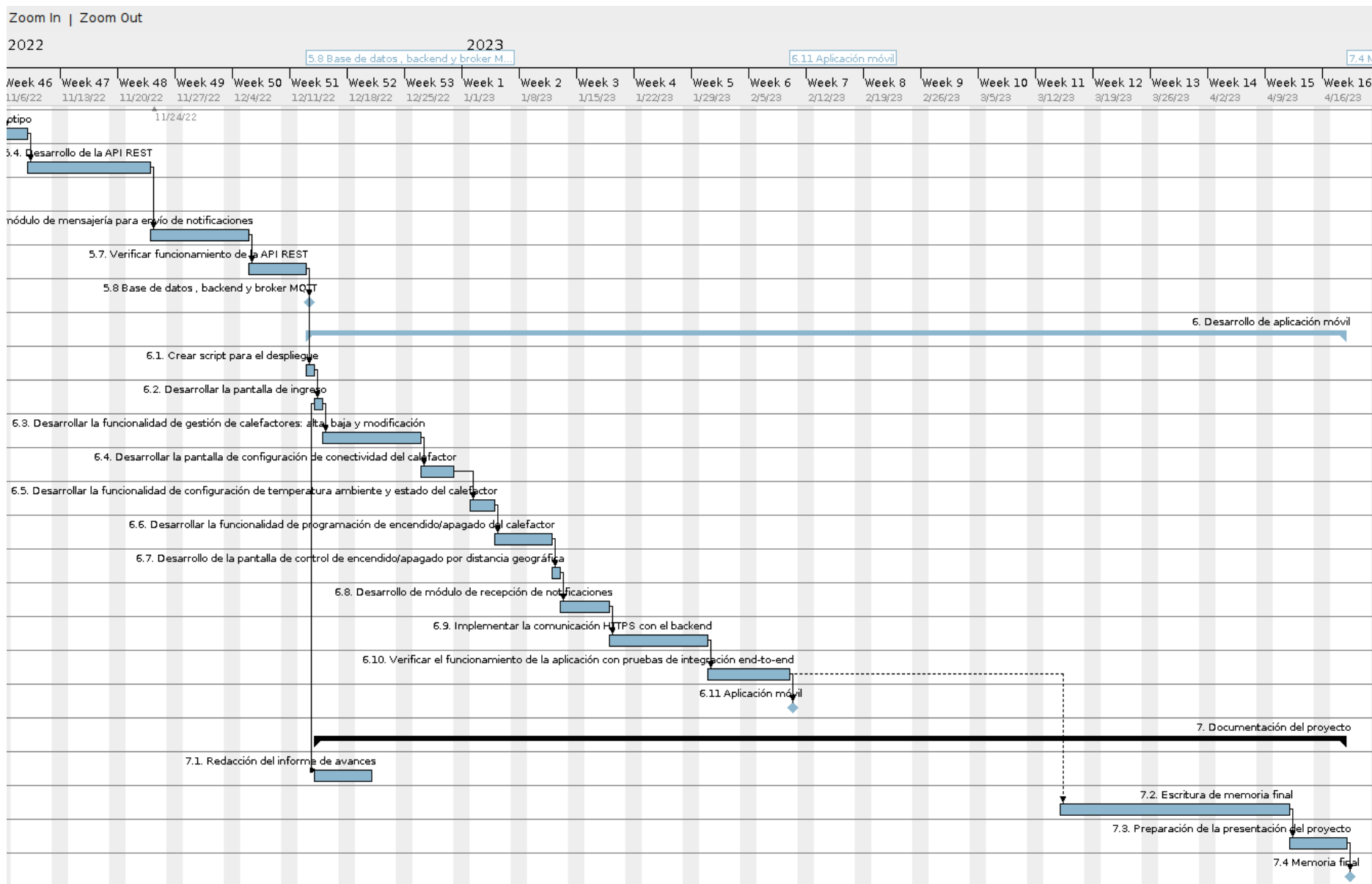


Figura 7. Diagrama de Gantt.

12. Presupuesto detallado del proyecto

A continuación se detallan los costos asociados el proyecto expresados en pesos argentinos (AR\$).

| COSTOS DIRECTOS | | | |
|---|----------|----------------|-------------|
| Descripción | Cantidad | Valor unitario | Valor total |
| Horas de trabajo | 657 | \$1500 | \$985500 |
| Módulo esp-32 | 1 | \$3000 | \$3000 |
| Hardware prototipo controlador del calefactor | 1 | \$20000 | \$20000 |
| SUBTOTAL | | | \$1008500 |
| COSTOS INDIRECTOS | | | |
| Descripción | Cantidad | Valor unitario | Valor total |
| Conexión mensual a Internet | 10 | \$4000 | \$40000 |
| Matricula posgrado | 1 | \$40000 | \$40000 |
| Cuota posgrado | 10 | \$16300 | \$163000 |
| 30 % sobre costos directos | 1 | \$302550 | \$302550 |
| SUBTOTAL | | | \$545550 |
| TOTAL | | | \$1554050 |

13. Gestión de riesgos

Riesgo 1: demora en obtener la placa controladora del calefactor.

- Severidad (6): la demora en obtener la placa controladora retrasaría el inicio del proyecto ya que es necesaria para las primeras tareas planificadas.
- Probabilidad de ocurrencia (2): probabilidad baja ya que se ha coordinado con el cliente el envío de la placa controladora para que esté disponible en las primeras semanas de junio.

Riesgo 2: aumento de costos de módulo Wi-Fi.

- Severidad (8): el módulo Wi-Fi es un componente importante en el proyecto, no poder adquirirlo provocaría tener que re definir la tecnología a utilizar.
- Ocurrencia (6): los precios de placas controladoras y componentes de hardware han tenido aumentos importantes en los últimos meses.

Riesgo 3: demora en implementar servicios cloud con la tecnología seleccionada.

- Severidad (6): la demora en implementar servicios REST y broker MQTT provocaría que el proyecto se extienda en el tiempo estipulado.
- Ocurrencia (5): al ser una tecnología nueva para el equipo de desarrollo es probable que se produzcan bloqueos por falta de conocimiento.

Riesgo 4: los requerimientos tienen una mayor complejidad que la esperada.

- Severidad (7): el proyecto demandaría mas tiempo del estimado.
- Ocurrencia (3): falta de conocimientos del tema seleccionado.

Riesgo 5: el cliente decide no continuar el proyecto.

- Severidad (9): la decisión del cliente de no continuar el proyecto por decisión unilateral provocaría que tenga que redefinirse el plan.
- Ocurrencia (2): el cliente está muy comprometido con el proyecto y los tiempos planificados son acordes a sus expectativas.

Tabla de gestión de riesgos: (El RPN se calcula como $RPN=SxO$)

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 30.

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

| Riesgo | S | O | RPN | S* | O* | RPN* |
|--|---|---|-----|----|----|------|
| 1. Demora en obtener placa | 6 | 2 | 12 | | | |
| 2. Aumento de costo del módulo Wi-Fi | 8 | 6 | 48 | 7 | 3 | 21 |
| 3. Demora en implementar servicios cloud | 6 | 5 | 30 | 4 | 3 | 12 |
| 4. Mayor complejidad en los requerimientos | 7 | 3 | 21 | | | |
| 5. Cliente decide no continuar | 9 | 2 | 18 | | | |

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

Riesgo 2: se cuenta con módulos Wi-Fi propios a modo de respaldo.

- Severidad (7): en caso de aumentar el precio se pueden utilizar los módulos ya adquiridos para otros proyectos. Se debe tener cuidado de no dañarlos para evitar tener que comprar nuevos módulos.
- Probabilidad de ocurrencia (3): es poco probable que se rompan los módulos asignados.

Riesgo 3: adquirir conocimientos de la tecnología a utilizar mediante los temas vistos en las materias de la especialidad. Realizar consultas de dudas con expertos en la materia.

- Severidad (4): se reducirá la demora en tener listos los servicios cloud al contar con los conocimientos técnicos y la posibilidad de realizar consultas técnicas.
- Probabilidad de ocurrencia (3): los bloqueos se van a reducir ya que en la etapa de construcción de servicios cloud ya se contará con los conocimientos necesarios.

14. Gestión de la calidad

- Requerimiento #1.1: el usuario podrá dar de alta, baja y modificar información (dirección, nombre y descripción) de los calefactores en el sistema.
 - Verificación: debe existir una pantalla en la aplicación que permita agregar, eliminar y modificar datos de calefactores.
 - Validación: realizar mediante la aplicación las operaciones y deberán poder verse en la pantalla.
- Requerimiento #1.2: el calefactor deberá medir la temperatura ambiente en grados centígrados y con una precisión de ± 0.5 ° C.
 - Verificación: revisar las especificaciones técnicas del sensor de medición de temperatura.
 - Validación: no aplica.
- Requerimiento #1.3: el usuario podrá establecer el valor de la temperatura del calefactor en grados centígrados.
 - Verificación: la aplicación debe tener incluida pantalla para establecer temperatura.
 - Validación: utilizando la aplicación configurar la temperatura del calefactor.
- Requerimiento #1.4: el usuario podrá configurar hora y fecha de encendido y apagado de cada calefactor.
 - Verificación: la aplicación debe tener incluida pantalla para la programación de encendido y apagado de calefactores.
 - Validación: utilizando la aplicación programar el encendido y apagado de un calefactor.
- Requerimiento #1.5: el usuario podrá configurar una distancia de apagado/encendido automático en un rango de 1-50 km.
 - Verificación: la aplicación debe tener incluida pantalla para configurar distancia/apagado por distancia. Los valores permitidos deben ser en el rango de 1 a 5 kms.
 - Validación: utilizando la aplicación configurar una distancia de encendido/apagado.
- Requerimiento #1.6: el usuario podrá establecer el valor de kWh (Kilo-watt hora).
 - Verificación: la aplicación debe tener incluida pantalla para establecer el valor de kWh (Kilo-watt hora).
 - Validación: utilizando la aplicación establecer el valor de kWh (Kilo-watt hora) y poder ver la configuración en la pantalla.
- Requerimiento #1.7: el usuario podrá consultar el consumo de energía en kWh en el mes en curso.

- Verificación: el servidor debe almacenar el tiempo de encendido de cada calefactor del usuario.
- Validación: utilizando la aplicación ver el consumo por pantalla.
- Requerimiento #1.8: se deberá registrar las coordenadas geográficas de cada calefactor en el sistema.
 - Verificación: la aplicación debe tener campos para ingresar las coordenadas geográficas de ubicación de un calefactor y deben almacenarse en la base de datos.
 - Validación: utilizando la aplicación ingresar coordenadas de latitud y longitud y poder verlas por pantalla.
- Requerimiento #2.1: se podrá configurar la red Wi-Fi.
 - Verificación: el firmware del módulo Wi-Fi debe permitir configurar SSID y password de red Wi-Fi. Configurar mediante comandos y verificar que se conecte a Internet.
 - Validación: utilizando la aplicación configurar el Wi-Fi del calefactor y comprobar que se conecte a Internet.
- Requerimiento #2.2: se podrá configurar el reloj interno.
 - Verificación: el firmware del módulo Wi-Fi debe tener lógica de sincronismo NTP. Monitorear por puerto serial la fecha sincronizada.
 - Validación: no aplica.
- Requerimiento #2.3: la comunicación con la electrónica del calefactor deberá ser serial.
 - Verificación: revisar esquemático de conexión entre modulo Wi-Fi y placa controladora. Observar por puerto serie los datos entregados por la placa controladora y compararlos con la trama esperada.
 - Validación: Enviarle al cliente la trama recibida de la placa controladora para que la valide.
- Requerimiento #2.4: deberá informar los datos de temperatura al servidor cada 10 segundos.
 - Verificación: el firmware del módulo Wi-Fi debe tener una tarea que se ejecute cada 10 segundos y envíe el valor de temperatura al servidor.
 - Validación: no aplica.
- Requerimiento #2.5: en caso de mal funcionamiento, no detectar temperatura, informará el error mediante MQTT al servidor.
 - Verificación: el firmware del módulo Wi-Fi debe tener lógica de evento de no detección de temperatura. Para simular evento se desconectará la conexión entre placa controladora y módulo Wi-Fi. Se verificará finalmente que en el servidor MQTT exista el evento.
 - Validación: no aplica.
- Requerimiento #3.1: el calefactor permitirá ser configurado mediante Wi-Fi Direct.

- Verificación: el módulo Wi-Fi debe permitir conexión Wi-Fi Direct. La aplicación deberá tener opción de conectarse mediante Wi-Fi Direct.
- Validación: desde la aplicación conectarse mediante Wi-Fi direct a un calefactor.
- Requerimiento #3.2: el calefactor podrá ser accedido mediante Internet para ser configurado y mostrar la temperatura actual.
 - Verificación: el módulo Wi-Fi pueda ser accedido desde Internet.
 - Validación: desde la aplicación conectarse mediante Internet a un calefactor.
- Requerimiento #3.3: el protocolo de comunicación entre servidor y calefactor deberá ser bi-direccional mediante MQTT.
 - Verificación: el módulo Wi-Fi debe tener lógica de conexión mediante MQTT, lógica de publicación y suscripción.
 - Validación: no aplica.
- Requerimiento #3.4: las rutas MQTT para identificar cada calefactor deberán seguir una jerarquía: /usuario/usuario/ubicacion/ubicacion/calefactor/calefactor.
 - Verificación: el módulo Wi-Fi debe enviar datos mediante MQTT a la ruta /usuario/usuario/ubicacion/ubicacion/calefactor/calefactor. Verificar en broker MQTT la existencia de la ruta una vez que el calefactor publique datos.
 - Validación: no aplica.
- Requerimiento #3.5: la comunicación entre el dispositivo móvil y el servidor deberá ser mediante APIs REST.
 - Verificación: el servidor cloud deberá incluir endpoints para la comunicación con la aplicación. Ejecutar requests con un cliente REST y verificar que los response sean los esperados por la aplicación.
 - Validación: no aplica.
- Requerimiento #3.6: el servidor, si está configurado el aviso por cercanía, deberá enviar el evento al calefactor.
 - Verificación: configurar un calefactor para que pueda ser encendido mediante evento de cercanía. Enviar mediante endpoint REST evento de encendido por cercanía. El servidor debe enviar mediante MQTT evento de encendido al calefactor.
 - Validación: no aplica.
- Requerimiento #4.1: la aplicación móvil deberá mostrar un listado de calefactores del usuario conectado.
 - Verificación: la aplicación deberá tener una pantalla donde se listen los calefactores asociados al usuario.
 - Validación: ingresar a la aplicación y dar de alta uno o varios calefactores. Luego validar que esos calefactores aparezcan en la pantalla de listado de calefactores.
- Requerimiento #4.2: la aplicación móvil mostrará la temperatura actual del ambiente y los valores de configuración del calefactor.

- Verificación: la aplicación deberá tener una pantalla donde se pueda consultar la temperatura ambiente y valores de configuración para un calefactor seleccionado. Verificar que los datos mostrados sean los mismos a los datos almacenados en la base de datos.
- Validación: ingresar a la aplicación y seleccionar un calefactor. Ver los datos del calefactor en la pantalla.
- Requerimiento #4.3: la aplicación móvil deberá permitir ingresar el costo del kWh en el sistema para calcular costos de consumo.
 - Verificación: la aplicación deberá tener una pantalla donde se pueda ingresar el costo en pesos del kWh.
 - Validación: ingresar a la pantalla de configuración de costo de kWh y establecer un valor. Ver en la pantalla mensaje de configuración exitosa.
- Requerimiento #4.4: la aplicación móvil deberá mostrar el consumo en kW y el costo en pesos acumulados en el mes actual.
 - Verificación: la aplicación debe tener una pantalla donde se pueda consultar el consumo total de un mes seleccionado y el costo asociado.
 - Validación: ingresar a la aplicación y en la pantalla de consultas seleccionar el mes en curso, se deberá ver por pantalla el valor calculado.
- Requerimiento #4.5: la aplicación móvil permitirá recibir y mostrar notificaciones cuando ocurre un evento con alguno de los calefactores.
 - Verificación: verificar que la aplicación tenga los permisos necesarios para recibir notificaciones. El servidor cloud deberá tener módulo de envío de notificaciones push cuando ocurran eventos.
 - Validación: generar un evento de error en el calefactor y en la aplicación del usuario deberá recibirse una notificación informando el evento.
- Requerimiento #4.6: la aplicación móvil deberá tener una pantalla de ingreso a la aplicación.
 - Verificación: verificar que la aplicación tenga pantalla para que el usuario ingrese. La pantalla debe tener campo de usuario y contraseña. Los datos de usuario y contraseña deben ser enviados al servidor y autorizar el ingreso en caso de ser correctos.
 - Validación: ingresar a la aplicación con datos de un usuario existente.
- Requerimiento #4.7: la aplicación móvil deberá capturar las coordenadas de gps con una frecuencia de 10 minutos.
 - Verificación: verificar que la aplicación tenga permisos para obtener coordenadas GPS. Debe existir lógica para capturar las coordenadas cada 10 minutos. Se utilizarán herramientas de debug para verificar que los datos sean capturados correctamente.
 - Validación: no aplica.
- Requerimiento #4.8: la aplicación móvil deberá calcular si se encuentra geográficamente en el rango de la ubicación del calefactor y en caso afirmativo enviar evento al servidor.

- Verificación: verificar que la aplicación tenga lógica de calcula de cercanía a los calefactores utilizando las coordenadas de GPS. Se utilizarán herramientas de debug para verificar que el calculo de cercanía se realice correctamente.
- Validación: dar de alta un calefactor en el sistema , luego habilitar apagado por cercanía con una distancia de 1 km. Utilizando la aplicación con el GPS activad ingresar en el radio del calefactor y observar que el calefactor se encienda correctamente.
- Requerimiento #5.1: el módulo de comunicaciones permitirá la configuración de red Wi-Fi mediante WPS.
 - Verificación: revisar especificaciones técnicas del módulo Wi-Fi.
 - Validación: no aplica.
- Requerimiento #5.2: el servidor, con el broker y su base de datos deberán estar alojados en un entorno cloud.
 - Verificación: verificar que el servidor, broker y base de datos estén correctamente instalados en el proveedor cloud seleccionado.
 - Validación: el cliente ingresa a los distintos componentes utilizando las credenciales suministradas.
- Requerimiento #5.3: los datos enviados por MQTT deberán estar estar cifrados con TLS..
 - Verificación: el firmware debe tener instalado los certificados necesarios para poder conectarse al servidor MQTT mediante TLS. En el servidor cloud deben estar instalados los certificados TLS y realizar una prueba de conexión. Se utilizarán certificados self-signed.
 - Validación: no aplica.
- Requerimiento #5.4: la comunicación con la API Rest deberá ser mediante HTTPS.
 - Verificación: el servidor debe tener los certificados instalados para comunicación HTTPS y no aceptar conexiones HTTP. La aplicación móvil debe tener el certificado instalado y tener la lógica de conexión https implementada. e utilizarán certificados self-signed.
 - Validación: no aplica.
- Requerimiento #5.5: la aplicación móvil deberá utilizar componentes de "Material Design".
 - Verificación: los componentes visuales de la aplicación móvil deben pertenecer a la librería "material design".
 - Validación: no aplica.

15. Procesos de cierre

- Pautas de trabajo que se seguirán para analizar si se respetó el Plan de Proyecto original
 - Encargado: Leonardo Mancini.
 - Se revisará el plan de trabajo analizando punto por punto el cumplimiento de los requerimientos y objetivos.
 - Se revisará junto con el diagrama de Gantt si hubo demoras, en caso de haberlas se analizarán los motivos.
 - Se medirá el grado de satisfacción con el cliente.
- Identificación de las técnicas y procedimientos útiles e inútiles que se emplearon, y los problemas que surgieron y cómo se solucionaron:
 - Encargado: Leonardo Mancini.
 - Se realizará una reunión con el cliente.
 - Se discutirán las técnicas de desarrollo de software utilizadas y cuales de estas agregaron mas valor al proyecto.
 - Se identificará las técnicas que se podrían incorporar al área de desarrollo del cliente.
 - Se analizaran los problemas que surgieron y como se resolvieron.
 - Se analizaran las malas practicas que se detectaron en el proyecto para evitar repetirlas en proyectos similares.
- Indicar quién organizará el acto de agradecimiento a todos los interesados:
 - Encargado: Leonardo Mancini.
 - Una vez finalizada la defensa pública del proyecto se procederá a agradecer a todos los involucrados: cliente, directores, autoridades de la carrera y compañeros.
 - Se enviará presentación del proyecto y memoria al cliente y colaboradores.