

## 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 . . . . .	7
7. Historias de usuarios ( <i>Product backlog</i> ). . . . .	8
7.1 Usuarios . . . . .	8
7.2 Sensores disponibles . . . . .	9
7.3 Mediciones de un sensor . . . . .	9
7.4 Alarmas . . . . .	9
7.5 Información de los sensores . . . . .	10
8. Entregables principales del proyecto . . . . .	10
9. Desglose del trabajo en tareas . . . . .	10

## 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 . . . . .	7
7. Historias de usuarios ( <i>Product backlog</i> ). . . . .	8
7.1 Usuarios . . . . .	8
7.2 Sensores disponibles . . . . .	9
7.3 Mediciones de un sensor . . . . .	9
7.4 Alarmas . . . . .	9
7.5 Información de los sensores . . . . .	10
8. Entregables principales del proyecto . . . . .	10
9. Desglose del trabajo en tareas . . . . .	10
10. Diagrama de Activity On Node. . . . .	12
11. Diagrama de Gantt. . . . .	12
12. Presupuesto detallado del proyecto . . . . .	14

### Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	18 de octubre de 2021
1	Se completa hasta el punto 5 inclusive	4 de noviembre de 2021
2	Se completa hasta el punto 9 inclusive y se realizan las correcciones de la entrega 1.	11 de noviembre de 2021

### Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	18 de octubre de 2021
1	Se completa hasta el punto 5 inclusive	4 de noviembre de 2021
2	Se completa hasta el punto 9 inclusive y se realizan las correcciones de la entrega 1.	11 de noviembre de 2021
3	Se completa hasta el punto 12 inclusive y se realizan las correcciones de la entrega 2.	18 de noviembre de 2021

## Acta de constitución del proyecto

Buenos Aires, 18 de octubre de 2021

Por medio de la presente se acuerda con el Ing. Funes Pablo Nicolás que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará "Plataforma multipropósito de adquisición y visualización", consistirá esencialmente en **la implementación de una plataforma genérica de adquisición y visualización**, y tendrá un presupuesto preliminar estimado de XXX hs de trabajo y \$XXX, con fecha de inicio 18 de octubre de 2021 y fecha de presentación pública 6 de diciembre de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg  
Director posgrado FIUBA

Funes Pablo Nicolás  
Desarrollo personal

Gustavo Zocco  
Director del Trabajo Final

## Acta de constitución del proyecto

Buenos Aires, 18 de octubre de 2021

Por medio de la presente se acuerda con el Ing. Funes Pablo Nicolás que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará "Plataforma multipropósito de adquisición y visualización", consistirá esencialmente en **la implementación de una plataforma genérica de adquisición y visualización**, y tendrá un presupuesto preliminar estimado de XXX hs de trabajo y \$XXX, con fecha de inicio 18 de octubre de 2021 y fecha de presentación pública 6 de diciembre de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg  
Director posgrado FIUBA

Funes Pablo Nicolás  
Desarrollo personal

Gustavo Zocco  
Director del Trabajo Final

## 1. Descripción técnica-conceptual del proyecto a realizar

Hoy en día los seres humanos nos encontramos constantemente rodeados de información, la información paso de ser un recurso escaso a un recurso abundante disponible para su explotación a partir de la era digital.

Debido a este gran volumen de información, se genera la necesidad de disponer algún medio de discriminación para la información útil. La información requiere técnicas de análisis y procesamiento de datos para ser discriminada.

La información se encuentra disponible para capitalizar en distintos sectores y procesos. A modo de ejemplo, se pueden mencionar una industria química, una papelería o incluso un estacionamiento. En todos los casos mencionados, disponemos de elementos capaces de recolectar la información, en el caso de la industria química podemos considerar sensores de pH, temperatura, posición y energía. De forma similar al ejemplo de la industria química, se podrían mencionar sensores para los ejemplos de la papelería o el estacionamiento.

Un sensor se puede definir como un dispositivo capaz de detectar y responder a algún tipo de entrada del entorno físico. Los tipos de entrada pueden ser luz, calor, movimiento, humedad, presión o cualquiera de un gran número de otros fenómenos ambientales. La salida es generalmente una señal que se convierte en una pantalla legible por humanos en la ubicación del sensor o se transmite electrónicamente a través de una red para su lectura o procesamiento adicional.

El avance de la tecnología permitió el surgimiento de una infinidad de sensores con distintas capacidades físicas, almacenamiento y comunicación. A partir de esta característica, se puede mencionar que existe una gran cantidad de información dispersa en distintos elementos independientes.

Hoy en día existen distintas aplicaciones que integran la información recopilada por los distintos sensores en una única plataforma, no obstante los costos de licencia, compatibilidad, usos de aplicación y el soporte son fundamentales para su funcionamiento.

Las grandes industrias no tienen problemas de recursos para afrontar dicho problema. Por el contrario pequeñas industrias, comercios, cooperativas entre otros, presentan recursos acotados y no disponen las capacidades para integrar dichos sistemas.

El presente proyecto se destaca especialmente por la generalidad del mismo, en lugar de buscar diferentes soluciones para distintos problemas, se propone una solución que se pueda adaptar a los distintos problemas y ambientes. En este trabajo se busca desarrollar una aplicación, la aplicación podrá ser utilizada en la mayor cantidad de escenarios posibles realizando configuraciones mínimas.

Con este enfoque se propone separar el desarrollo en dos capas, una capa funcional común para cualquier problema y una capa de desarrollo particular para el cliente en cuestión.

En la figura 1 se presenta el diagrama en bloques del sistema, se puede observar por un lado los clientes mobile/web y por otro lado el backend. Dentro del bloque de backend se puede distinguir los bloques denominados core genérico y workers. El bloque core genérico corresponde a la capa funcional común para los problemas como se menciono previamente, mientras que el bloque workers por su lado se corresponde con la lógica particular de los distintos problemas.

A su vez, en la figura 1 se puede observar la existencia de un backoffice para realizar las configuraciones correspondientes para un cliente determinado desde un navegador web.

## 1. Descripción técnica-conceptual del proyecto a realizar

Hoy en día los seres humanos nos encontramos constantemente rodeados de información, la información paso de ser un recurso escaso a un recurso abundante disponible para su explotación a partir de la era digital.

Debido a este gran volumen de información, se genera la necesidad de disponer algún medio de discriminación para la información útil. La información requiere técnicas de análisis y procesamiento de datos para ser discriminada.

La información se encuentra disponible para capitalizar en distintos sectores y procesos. A modo de ejemplo, se pueden mencionar una industria química, una papelería o incluso un estacionamiento. En todos los casos mencionados, disponemos de elementos capaces de recolectar la información, en el caso de la industria química podemos considerar sensores de pH, temperatura, posición y energía. De forma similar al ejemplo de la industria química, se podrían mencionar sensores para los ejemplos de la papelería o el estacionamiento.

Un sensor se puede definir como un dispositivo capaz de detectar y responder a algún tipo de entrada del entorno físico. Los tipos de entrada pueden ser luz, calor, movimiento, humedad, presión o cualquiera de un gran número de otros fenómenos ambientales. La salida es generalmente una señal que se convierte en una pantalla legible por humanos en la ubicación del sensor o se transmite electrónicamente a través de una red para su lectura o procesamiento adicional.

El avance de la tecnología permitió el surgimiento de una infinidad de sensores con distintas capacidades físicas, almacenamiento y comunicación. A partir de esta característica, se puede mencionar que existe una gran cantidad de información dispersa en distintos elementos independientes.

Hoy en día existen distintas aplicaciones que integran la información recopilada por los distintos sensores en una única plataforma, no obstante los costos de licencia, compatibilidad, usos de aplicación y el soporte son fundamentales para su funcionamiento.

Las grandes industrias no tienen problemas de recursos para afrontar dicho problema. Por el contrario pequeñas industrias, comercios, cooperativas entre otros, presentan recursos acotados y no disponen las capacidades para integrar dichos sistemas.

El presente proyecto se destaca especialmente por la generalidad del mismo, en lugar de buscar diferentes soluciones para distintos problemas, se propone una solución que se pueda adaptar a los distintos problemas y ambientes. En este trabajo se busca desarrollar una aplicación, la aplicación podrá ser utilizada en la mayor cantidad de escenarios posibles realizando configuraciones mínimas.

Con este enfoque se propone separar el desarrollo en dos capas, una capa funcional común para cualquier problema y una capa de desarrollo particular para el cliente en cuestión.

En la figura 1 se presenta el diagrama en bloques del sistema, se puede observar por un lado los clientes mobile/web y por otro lado el backend. Dentro del bloque de backend se puede distinguir los bloques denominados core genérico y workers. El bloque core genérico corresponde a la capa funcional común para los problemas como se menciono previamente, mientras que el bloque workers por su lado se corresponde con la lógica particular de los distintos problemas.

A su vez, en la figura 1 se puede observar la existencia de un backoffice para realizar las configuraciones correspondientes para un cliente determinado desde un navegador web.

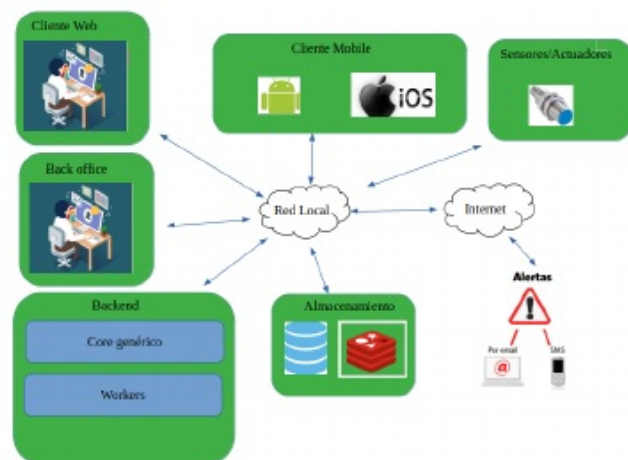


Figura 1. Diagrama en bloques del sistema

## 2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Pablo Valentín Funes	Funes - Cerial Consultores en Ingeniería	Ingeniero eléctrico
Responsable	Funes Pablo Nicolás	FIUBA	Alumno
Orientador	Gustavo Zocco	FIUBA	Director Trabajo final

## 3. Propósito del proyecto

El propósito de este proyecto es generar una plataforma IOT multipropósito. La plataforma servirá de base para futuros trabajos de ingeniería, proporcionándole al alumno los conocimientos básicos para desarrollarse en los proyectos que puedan surgir.

Estos trabajos consistirían en la configuración de la plataforma para el cliente, con la posibilidad de integrar los distintos equipos en la plataforma como un servicio de ingeniería o un servicio de inteligencia y análisis de datos.

## 4. Alcance del proyecto

El proyecto incluye el desarrollo de un sistema encargado de:

1. Proveer una interfaz de usuario mobile (ios/android).
2. Proveer una interfaz de usuario web.

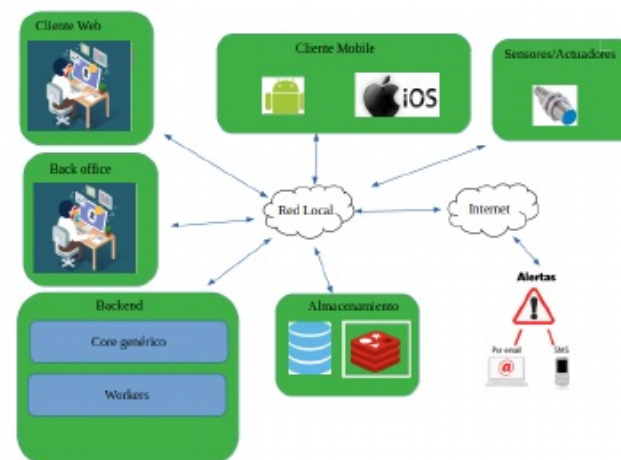


Figura 1. Diagrama en bloques del sistema

## 2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Pablo Valentín Funes	Funes - Cerial Consultores en Ingeniería	Ingeniero eléctrico
Responsable	Funes Pablo Nicolás	FIUBA	Alumno
Orientador	Gustavo Zocco	FIUBA	Director Trabajo final

## 3. Propósito del proyecto

El propósito de este proyecto es generar una plataforma IOT multipropósito. La plataforma servirá de base para futuros trabajos de ingeniería, proporcionándole al alumno los conocimientos básicos para desarrollarse en los proyectos que puedan surgir.

Estos trabajos consistirían en la configuración de la plataforma para el cliente, con la posibilidad de integrar los distintos equipos en la plataforma como un servicio de ingeniería o un servicio de inteligencia y análisis de datos.

## 4. Alcance del proyecto

El proyecto incluye el desarrollo de un sistema encargado de:

1. Proveer una interfaz de usuario mobile (ios/android).
2. Proveer una interfaz de usuario web.

3. Proveer una interfaz de configuración web back office para administrar las funcionalidades según el cliente.
4. Registrar de forma persistente la información obtenida.
5. Generar alertas según parámetros establecidos.
6. Registrar las mediciones realizadas por sensores y actuadores.
7. Desarrollo del firmware en nodos sensores/actuadores para banco de pruebas.
8. Desarrollo del hardware en nodos sensores/actuadores para banco de pruebas.
9. Pruebas sobre la plataforma utilizando sensores de temperatura/humedad DHT22 como variables de entrada y 2 leds cumpliendo el rol de variables de salida.

El proyecto no incluye:

1. Contratación de servicios de terceros.
2. Desarrollo de hardware de adquisición de datos.
3. Desarrollo de firmware para el hardware de adquisición de datos.
4. Desarrollos para las plataformas de Apple.

## 5. Supuestos del proyecto

Para el desarrollo del presente proyecto se realizan los siguientes supuestos:

- Se dispondrán los recursos económicos suficientes para adquirir los componentes necesarios para el banco de pruebas.
- Se dispondrá de los recursos necesarios para realizar la investigación de los recursos a emplear durante el proyecto.
- Se dispondrá el apoyo de los orientadores a lo largo del proyecto.
- Se dispondrá un alcance de las funcionalidades acorde al tiempo estipulado del proyecto (1 año).

## 6. Requerimientos

A continuación se muestran los requerimientos del proyecto clasificados según el componente que tengan en común y la prioridad. A menor valor numérico, la prioridad es mayor.

1. Requerimientos interfaz cliente (web-mobile)
  - 1.1. El sistema debe funcionar en los clientes web chrome y mozilla.
  - 1.2. El sistema debe funcionar en ios/android.

3. Proveer una interfaz de configuración web back office para administrar las funcionalidades según el cliente.
4. Registrar de forma persistente la información obtenida.
5. Generar alertas según parámetros establecidos.
6. Registrar las mediciones realizadas por sensores y actuadores.
7. Desarrollo del firmware en nodos sensores/actuadores para banco de pruebas.
8. Desarrollo del hardware en nodos sensores/actuadores para banco de pruebas.
9. Pruebas sobre la plataforma utilizando sensores de temperatura/humedad DHT22 como variables de entrada y 2 leds cumpliendo el rol de variables de salida.

El proyecto no incluye:

1. Contratación de servicios de terceros.
2. Desarrollo de hardware de adquisición de datos.
3. Desarrollo de firmware para el hardware de adquisición de datos.
4. Desarrollos para las plataformas de Apple.

## 5. Supuestos del proyecto

Para el desarrollo del presente proyecto se realizan los siguientes supuestos:

- Se dispondrán los recursos económicos suficientes para adquirir los componentes necesarios para el banco de pruebas.
- Se dispondrá de los recursos necesarios para realizar la investigación de los recursos a emplear durante el proyecto.
- Se dispondrá el apoyo de los orientadores a lo largo del proyecto.
- Se dispondrá un alcance de las funcionalidades acorde al tiempo estipulado del proyecto (1 año).

## 6. Requerimientos

A continuación se muestran los requerimientos del proyecto clasificados según el componente que tengan en común y la prioridad. A menor valor numérico, la prioridad es mayor.

1. Requerimientos interfaz cliente (web-mobile)
  - 1.1. El sistema debe funcionar en los clientes web chrome y mozilla.
  - 1.2. El sistema debe funcionar en ios/android.

- 1.3. El usuario debe poder listar todos los sensores disponibles.
- 1.4. El usuario debe poder visualizar las mediciones de los sensores disponibles.
- 1.5. El usuario debe poder accionar un actuador en caso de contar con los permisos.
2. Requerimientos del backoffice
  - 2.1. El usuario debe poder ingresar mediante un login.
  - 2.2. El sistema debe permitir la configuración de las funcionalidades disponibles.
  - 2.3. El sistema debe permitir la actualización de los parámetros de un usuario.
  - 2.4. El sistema debe permitir la actualización de los parámetros de un sensor.
  - 2.5. El sistema debe permitir la configuración de una alerta a partir de los parámetros de un sensor.
  - 2.6. Requerimiento 2 (prioridad menor)
3. Requerimientos del backend
  - 3.1. El sistema debe persistir la información de los usuarios.
  - 3.2. El sistema debe persistir la información de los sensores.
  - 3.3. El sistema debe poder informar con una alerta si es necesario.
  - 3.4. El sistema debe poder comunicarse con los protocolos http-mqtt.
4. Requerimiento de testing.
  - 4.1. El sistema debe disponer una suite de pruebas unitarias para asegurar el funcionamiento.
  - 4.2. El sistema debe disponer un banco de pruebas con sensores y actuadores.
  - 4.3. El sistema debe disponer una serie de pruebas de integración utilizando el banco de pruebas.

## 7. Historias de usuarios (*Product backlog*)

En esta sección se incluyen las historias de usuarios y sus correspondientes ponderaciones.

### 7.1. Usuarios

Como usuario quiero registrarme en la plataforma e ingresar con mi correo electrónico para garantizar la seguridad de mi información.

- Dificultad: 3 - El registro del usuario involucra muchas horas para asegurar la autenticación del mismo.
- Complejidad: 8 - La integración con algún servicio externo puede volverse difícil y deben considerarse muchos casos borde.
- Riesgo: 5 - Se debe asegurar el mecanismo para corroborar la identidad al momento del registro del correo electrónico, generalmente se usan sms o email.

Story Points: 16

- 1.3. El usuario debe poder listar todos los sensores disponibles.
- 1.4. El usuario debe poder visualizar las mediciones de los sensores disponibles.
- 1.5. El usuario debe poder accionar un actuador en caso de contar con los permisos.
2. Requerimientos del backoffice
  - 2.1. El usuario debe poder ingresar mediante un login.
  - 2.2. El sistema debe permitir la configuración de las funcionalidades disponibles.
  - 2.3. El sistema debe permitir la actualización de los parámetros de un usuario.
  - 2.4. El sistema debe permitir la actualización de los parámetros de un sensor.
  - 2.5. El sistema debe permitir la configuración de una alerta a partir de los parámetros de un sensor.
  - 2.6. Requerimiento 2 (prioridad menor)
3. Requerimientos del backend
  - 3.1. El sistema debe persistir la información de los usuarios.
  - 3.2. El sistema debe persistir la información de los sensores.
  - 3.3. El sistema debe poder informar con una alerta si es necesario.
  - 3.4. El sistema debe poder comunicarse con los protocolos http-mqtt.
4. Requerimiento de testing.
  - 4.1. El sistema debe disponer una suite de pruebas unitarias para asegurar el funcionamiento.
  - 4.2. El sistema debe disponer un banco de pruebas con sensores y actuadores.
  - 4.3. El sistema debe disponer una serie de pruebas de integración utilizando el banco de pruebas.

## 7. Historias de usuarios (*Product backlog*)

En esta sección se incluyen las historias de usuarios y sus correspondientes ponderaciones.

### 7.1. Usuarios

Como usuario quiero registrarme en la plataforma e ingresar con mi correo electrónico para garantizar la seguridad de mi información.

- Dificultad: 3 - El registro del usuario involucra muchas horas para asegurar la autenticación del mismo.
- Complejidad: 8 - La integración con algún servicio externo puede volverse difícil y deben considerarse muchos casos borde.
- Riesgo: 5 - Se debe asegurar el mecanismo para corroborar la identidad al momento del registro del correo electrónico, generalmente se usan sms o email.

Story Points: 16



## 7.2. Sensores disponibles

Como usuario quiero poder visualizar el listado completo de los sensores disponibles para poder seleccionar el sensor de interés.

- Dificultad: 3 - Involucra hacer el pedido de información al backend y seleccionar la forma de mostrar la información.
- Complejidad: 5 - Hacer un servicio que devuelva información contemplando todos los casos posibles.
- Riesgo: 2 - Se puede dar el caso de no obtener información con el servicio teniendo que manejar un caso excepcional.

Story Points: 10

## 7.3. Mediciones de un sensor

Como usuario quiero poder visualizar las mediciones de un sensor seleccionado para observar el correcto funcionamiento del sistema.

- Dificultad: 3 - Involucra el pedido de información al backend y seleccionar la forma de mostrar la información.
- Complejidad: 5 - Hacer un servicio que devuelva información contemplando todos los casos posibles.
- Riesgo: 3 - Se puede dar el caso de no obtener información con el servicio teniendo que manejar un caso excepcional.

Story Points: 11

## 7.4. Alarmas

Como usuario quiero recibir notificaciones por la aplicación o correo electrónico en el caso de una alarma para conocer el estado actual del sistema.

- Dificultad: 5 - Se requiere correr un proceso de fondo revisando la situación particular de cada sensor para accionar una alarma.
- Complejidad: 13 - Cada sensor es diferente por lo que cada alarma es diferente y requiere darle un comportamiento particular.
- Riesgo: 5 - Puedo que no se dispare una alarma y ocurra un problema.

Story Points: 23

## 7.2. Sensores disponibles

Como usuario quiero poder visualizar el listado completo de los sensores disponibles para poder seleccionar el sensor de interés.

- Dificultad: 3 - Involucra hacer el pedido de información al backend y seleccionar la forma de mostrar la información.
- Complejidad: 5 - Hacer un servicio que devuelva información contemplando todos los casos posibles.
- Riesgo: 2 - Se puede dar el caso de no obtener información con el servicio teniendo que manejar un caso excepcional.

Story Points: 10

## 7.3. Mediciones de un sensor

Como usuario quiero poder visualizar las mediciones de un sensor seleccionado para observar el correcto funcionamiento del sistema.

- Dificultad: 3 - Involucra el pedido de información al backend y seleccionar la forma de mostrar la información.
- Complejidad: 5 - Hacer un servicio que devuelva información contemplando todos los casos posibles.
- Riesgo: 3 - Se puede dar el caso de no obtener información con el servicio teniendo que manejar un caso excepcional.

Story Points: 11

## 7.4. Alarmas

Como usuario quiero recibir notificaciones por la aplicación o correo electrónico en el caso de una alarma para conocer el estado actual del sistema.

- Dificultad: 5 - Se requiere correr un proceso de fondo revisando la situación particular de cada sensor para accionar una alarma.
- Complejidad: 13 - Cada sensor es diferente por lo que cada alarma es diferente y requiere darle un comportamiento particular.
- Riesgo: 5 - Puedo que no se dispare una alarma y ocurra un problema.

Story Points: 23



### 7.5. Información de los sensores

Como usuario quiero recibir la información de los sensores por correo electrónico en formato csv para poder realizar un posterior análisis de datos.

- Dificultad: 5 - Requiere la búsqueda de la información particular y el formato.
- Complejidad: 5 - Requiere enviar un archivo por un servicio externo.
- Riesgo: 3 - El archivo puede contener errores de formato y no enviarse.

Story Points: 13

### 8. Entregables principales del proyecto

Los principales entregables del proyecto son:

- Manual de uso
- Video tutorial de uso.
- Software ejecutable de la aplicación.
- Informe de avance del proyecto.
- Memoria del proyecto.

### 9. Desglose del trabajo en tareas

En esta sección se muestra el listado de tareas que forman parte del proyecto.

1. Planificación del proyecto (70 hs)
  - 1.1. Realizar el plan del proyecto (40 hs)
  - 1.2. Realizar el estudio de viabilidad económica-financiera del proyecto (20 hs)
  - 1.3. Aprobación de la planificación(10 hs)
2. Investigación y selección de tecnologías. (56 hs)
  - 2.1. Investigación y selección de las tecnologías en el desarrollo del software (32 hs)
  - 2.2. Investigación de protocolos http-mqtt (8 hs)
  - 2.3. Investigación sobre herramientas de deploy(16 hs)
3. Configuración de la infraestructura del proyecto. (58 hs)
  - 3.1. Configuración de las bases de datos (8 hs)
  - 3.2. Configuración de las instancias de backend (12 hs)
  - 3.3. Configuración de las instancias de frontend web (16 hs)

### 7.5. Información de los sensores

Como usuario quiero recibir la información de los sensores por correo electrónico en formato csv para poder realizar un posterior análisis de datos.

- Dificultad: 5 - Requiere la búsqueda de la información particular y el formato.
- Complejidad: 5 - Requiere enviar un archivo por un servicio externo.
- Riesgo: 3 - El archivo puede contener errores de formato y no enviarse.

Story Points: 13

### 8. Entregables principales del proyecto

Los principales entregables del proyecto son:

- Manual de uso
- Video tutorial de uso.
- Software ejecutable de la aplicación.
- Informe de avance del proyecto.
- Memoria del proyecto.

### 9. Desglose del trabajo en tareas

En esta sección se muestra el listado de tareas que forman parte del proyecto.

1. Planificación del proyecto (70 hs)
  - 1.1. Realizar el plan del proyecto (40 hs)
  - 1.2. Realizar el estudio de viabilidad económica-financiera del proyecto (20 hs)
  - 1.3. Aprobación de la planificación(10 hs)
2. Investigación y selección de tecnologías. (56 hs)
  - 2.1. Investigación y selección de las tecnologías en el desarrollo del software (32 hs)
  - 2.2. Investigación de protocolos http-mqtt (8 hs)
  - 2.3. Investigación sobre herramientas de deploy(16 hs)
3. Configuración de la infraestructura del proyecto. (58 hs)
  - 3.1. Configuración de las bases de datos (8 hs)
  - 3.2. Configuración de las instancias de backend (12 hs)
  - 3.3. Configuración de las instancias de frontend web (16 hs)

- 3.4. Configuración de un servicio externo para las pruebas unitarias CircleCi o Travis (6 hs).
- 3.5. Configuración de herramientas para el deploy de la aplicación (16 hs).
- 4. Desarrollo del backend. (72 hs)
  - 4.1. Desarrollo del endpoint del listado de sensores (8 hrs)
  - 4.2. Desarrollo del endpoint de las mediciones de un sensor (8 hs)
  - 4.3. Desarrollo del endpoint para el registro de un usuario (12 hs)
  - 4.4. Desarrollo del endpoint para el login de un usuario (8 hs)
  - 4.5. Desarrollo del backoffice (16 hs)
  - 4.6. Desarrollo del software para comunicacion con los sensores (20 hs)
- 5. Desarrollo del frontend web. (70 hs)
  - 5.1. Diseño de las vistas de la aplicación (32 hs)
  - 5.2. Desarrollo de la vista del login de un usuario (12 hs)
  - 5.3. Desarrollo de la vista de un menú principal con las funcionalidades (8 hs)
  - 5.4. Desarrollo de la vista del listado de sensores (12 hs)
  - 5.5. Desarrollo de la vista de las mediciones de un sensor (12 hs)
- 6. Desarrollo del frontend mobile. (70 hs)
  - 6.1. Diseño de las vistas de la aplicación (32)
  - 6.2. Desarrollo de la vista del login de un usuario (12 hs)
  - 6.3. Desarrollo de la vista de un menú principal con las funcionalidades (8 hs)
  - 6.4. Desarrollo de la vista del listado de sensores (12 hs)
  - 6.5. Desarrollo de la vista de las mediciones de un sensor (12 hs)
- 7. Testing. (60 hs)
  - 7.1. Diseño de casos de prueba (8 hs)
  - 7.2. Generación de datos de prueba (4 hs)
  - 7.3. Diseño del hardware de pruebas (8 hs)
  - 7.4. Ejecución del esquema de pruebas (34 hs)
  - 7.5. Generar reporte de las pruebas (6 hs)
- 8. Validación. (40 hs)
  - 8.1. Diseño de ensayos de validación (12 hrs)
  - 8.2. Ensayos de validación (28 hs)
- 9. Presentación del trabajo. (142 hs)
  - 9.1. Redacción del informe de avance (10 hs)
  - 9.2. Redacción del tutorial de uso (8 hs)
  - 9.3. Redacción de la memoria escrita (100 hs)
  - 9.4. Elaboración de presentación final (24 hs).

El tiempo total estimado del proyecto es de 638 hs.

- 3.4. Configuración de un servicio externo para las pruebas unitarias CircleCi o Travis (6 hs).
- 3.5. Configuración de herramientas para el deploy de la aplicación (16 hs).
- 4. Desarrollo del backend. (72 hs)
  - 4.1. Desarrollo del endpoint del listado de sensores (8 hrs)
  - 4.2. Desarrollo del endpoint de las mediciones de un sensor (8 hs)
  - 4.3. Desarrollo del endpoint para el registro de un usuario (12 hs)
  - 4.4. Desarrollo del endpoint para el login de un usuario (8 hs)
  - 4.5. Desarrollo del backoffice (16 hs)
  - 4.6. Desarrollo del software para comunicacion con los sensores (20 hs)
- 5. Desarrollo del frontend web. (70 hs)
  - 5.1. Diseño de las vistas de la aplicación (32 hs)
  - 5.2. Desarrollo de la vista del login de un usuario (12 hs)
  - 5.3. Desarrollo de la vista de un menú principal con las funcionalidades (8 hs)
  - 5.4. Desarrollo de la vista del listado de sensores (12 hs)
  - 5.5. Desarrollo de la vista de las mediciones de un sensor (12 hs)
- 6. Desarrollo del frontend mobile. (70 hs)
  - 6.1. Diseño de las vistas de la aplicación (32 hs)
  - 6.2. Desarrollo de la vista del login de un usuario (12 hs)
  - 6.3. Desarrollo de la vista de un menú principal con las funcionalidades (8 hs)
  - 6.4. Desarrollo de la vista del listado de sensores (12 hs)
  - 6.5. Desarrollo de la vista de las mediciones de un sensor (12 hs)
- 7. Testing. (60 hs)
  - 7.1. Diseño de casos de prueba (8 hs)
  - 7.2. Generación de datos de prueba (4 hs)
  - 7.3. Diseño del hardware de pruebas (8 hs)
  - 7.4. Ejecución del esquema de pruebas (34 hs)
  - 7.5. Generar reporte de las pruebas (6 hs)
- 8. Validación. (40 hs)
  - 8.1. Diseño de ensayos de validación (12 hrs)
  - 8.2. Ensayos de validación (28 hs)
- 9. Presentación del trabajo. (142 hs)
  - 9.1. Redacción del informe de avance (10 hs)
  - 9.2. Redacción del tutorial de uso (8 hs)
  - 9.3. Redacción de la memoria escrita (100 hs)
  - 9.4. Elaboración de presentación final (24 hs).

El tiempo total estimado del proyecto es de 638 hs.