

**GRADO EN INGENIERÍA INFORMÁTICA**

**ESPECIALIDAD EN TECNOLOGÍAS DE LA INFORMACIÓN**

Desarrollo de una infraestructura experimental para redes LoRa en malla

**TRABAJO DE FINAL DE GRADO**

**ENTREGABLE 1: CONTEXTUALIZACIÓN Y ALCANCE**

**JUAN CARLOS RUBIO DIAZ**

Director: Felix Freitag

Codirector: Roger Baig Viñas

Tutor GEP: Joan Subirats Soler

Semestre de otoño

2024-2025

**Tabla de contenidos**

[**1. Introducción y Contexto 4**](#_lf2v4mop32z7)

[1.1 Contextualización 5](#_jfp8xhyp94iw)

[1.2 Definición de conceptos 5](#_kt6jenmq6qht)

[1.2.1 LoRa 5](#_7fnnhf8wabc4)

[1.2.2 LoRa Mesher & LoRaChat 6](#_oaqds02t2uwg)

[**2. Alcance 8**](#_iv3y4vw9v3hc)

[2.1 Objetivo y propuestas 8](#_vlpcgpa2101s)

[2.2 Requisitos 9](#_lq8wl34j6o3c)

[2.3 Riesgos y obstáculos 9](#_gmdmrrrkzrmp)

[2.4 Metodología 10](#_q6u4wseylc3s)

[**3. Referencias 11**](#_5pkqlk2cirs)

**Lista de Figuras**

[**Figura 1: LoRaChat, proyecto en Visual Studio. 7**](#_fp0mpkl661mz)

# 1. Introducción y Contexto

En los últimos años, las tecnologías de comunicación inalámbrica han tenido un notable crecimiento debido a la expansión del Internet de las Cosas (o abreviando, IoT, por sus siglas en inglés) y la necesidad de conectar dispositivos de forma eficiente en todo tipo de entornos. Una de las tecnologías protagonistas en este contexto es LoRa (Long Range), un protocolo de comunicación que ofrece un largo alcance, usando baja potencia, por lo que es ideal para aplicarlo en este campo.

Sin embargo, no todo son ventajas. LoRa presenta limitaciones tales como la dependencia de una infraestructura de red centralizada basada en estaciones base (gateways) que no tiene porqué ser siempre adecuada. Con el fin de remediar esto, surgen las redes en malla (mesh networks) como alternativa para ampliar las capacidades de las redes LoRa, ya que, proporcionan una topología descentralizada donde cada nodo puede hacer de repetidor. Esto permite extender el alcance de la propia red y mejorar la resiliencia ante fallos de los nodos.

La descripción del proyecto era: “We have been building at UPC a library called LoRaMesher that enables creating LoRa mesh network with embedded devices. The project should make a contribution to this open source library and help to make it more usable or showing new usages[[1]](#footnote-0).”

El objetivo del trabajo es mejorar un trabajo ya existente. El punto de partida es el TFG de Jack Griffiths, también estudiante de la FIB, que conectó las placas que usaré en este proyecto. Por lo que respecta a nivel de madurez tecnológica o TRL, podríamos clasificar lo de Jack como TRL4 (nivel más bajo de simulación). En este trabajo se busca incrementar este TRL, teniendo como objetivo llegar a TRL6, el nivel más alto en la fase de simulación[[2]](#footnote-1).

## 1.1 Contextualización

Este trabajo corresponde al proyecto final del Grado en Ingeniería Informática con especialización en Tecnologías de la Información. El grado es cursado en la Facultad de Informática de Barcelona, dentro de la Universitat Politècnica de Catalunya.

El proyecto se realiza en modalidad A, publicado en la web del Racó de la Fib, y tiene como director a Felix Freitag y como codirector a Roger Baig Viñas.

Por lo que respecta a las competencias técnicas asociadas a este proyecto tenemos:[[3]](#footnote-2)

* CTI3.1: Concebir sistemas, aplicaciones y servicios basados en tecnologías de red, teniendo en cuenta Internet, web, comercio electrónico, multimedia, servicios interactivos y computación ubicua. [En profundidad].
* CTI3.3: Diseñar, implantar y configurar redes y servicios. [En profundidad].
* CTI3.4: Diseñar software de comunicaciones. [Bastante].

## 1.2 Definición de conceptos

### 1.2.1 LoRa

La tecnología LoRa, desarrollada por Semtech, se implementa dentro de una red LoRaWAN, en la cual dispositivos de IoT, como medidores inteligentes o sistemas de alarmas, se conectan a través de gateways que, a su vez, están conectados a Internet. Esto permite que los nodos de IoT interactúen con un servidor de red central. LoRaWAN es una red de área amplia de largo alcance (Wide Area Network), que usa los dispositivos IoT de LoRa, por lo que se ha convertido en la tecnología LPWAN más destacada y extendida. LoRa es preferida sobre otras soluciones debido a su bajo consumo de energía, mayor autonomía de batería, amplio rango de cobertura, coste reducido y facilidad de integración.

Actualmente, podemos encontrar LoRa en aplicaciones Smart, es decir, ciudades, hogares, agricultura y sistemas de medición inteligentes, ya que están impulsados por los propios dispositivos IoT.

Encontramos tres tipos de dispositivos LoRa dependiendo de su comportamiento y consumo:[[4]](#footnote-3)

* Clase A: Estos dispositivos son los más eficientes en cuanto a consumo energético, ya que permanecen inactivos la mayor parte del tiempo. Estos se activan para enviar un mensaje y entran en modo escucha hasta recibir una respuesta.
* Clase B: Consumen más energía que los de Clase A al activarse de manera periódica para enviar datos.
* Clase C: Los clase C están activos siempre en espera de recibir un mensaje, y cuando es necesario, transmiten. Esto los hace los dispositivos con mayor coste energético.

La decisión de utilizar LoRa en este proyecto se debe a que pretende ampliar una librería existente llamada LoRaMesher, que se detalla más adelante.

### 1.2.2 LoRa Mesher & LoRaChat

LoRaMesher es una librería desarrollada por Joan Miquel Solé, exalumno de la UPC, que implementa un protocolo de enrutamiento de vector de distancia en redes LoRa. Esta librería será utilizada en el proyecto para explorar y crear aplicaciones “cross-networks” en redes de malla LoRa.

En la página siguiente, se presenta una captura del código de LoRaChat, que utiliza LoRaMesher como servicio. LoRaChat facilita la creación de aplicaciones para dispositivos LoRa y será una herramienta clave en este proyecto para desarrollar las aplicaciones IoT deseadas. El proyecto está organizado de la siguiente forma:[[5]](#footnote-4)

* /src: Contiene las aplicaciones creadas para los dispositivos LoRa. Actualmente, ya existen varias apps con diferentes funciones, y el objetivo del proyecto es expandir esta carpeta con nuevas aplicaciones. Para ello, se utilizarán apps existentes como referencia.
* /src/message/dataMessage.h: Las aplicaciones deben estar referenciadas en este archivo para ser utilizadas (cada una está asignada a un número).
* config.h: Especifica la configuración de los nodos LoRa y las aplicaciones correspondientes, que pueden ser activadas o desactivadas.
* main.cpp: Aquí se crean las instancias de las aplicaciones para que puedan ser ejecutadas.
* platformio.ini: Archivo de configuración de PlatformIO. Define dispositivos IoT y librerías utilizadas.
* echo monitoring and cdp min.py: Archivo Python que actúa como publicador y suscriptor de los mensajes enviados a través de MQTT.

###### Figura 1: LoRaChat, captura del proyecto en Visual Studio.

###### 

###### 

###### 

###### 



# 2. Alcance

El proyecto se va a desarrollar siguiendo unos objetivos propuestos. Se hará el trabajo de investigación para ver si las ideas pensadas son viables en el mundo real y se justificará por qué sí, o, por que no valen para lo que se había imaginado inicialmente.

## 2.1 Objetivo y propuestas

El objetivo principal de este proyecto es contribuir al mundo del IoT mediante la creación de aplicaciones útiles para redes de malla LoRa. Otra forma de decirlo sería hacer un uso más común y extendido de este tipo de redes, haciendo que sean tanto eficientes como útiles.

Para conseguir este objetivo principal, hemos de alcanzar, también, los siguientes subobjetivos:

* Comprensión adecuada de las redes LoRa.
* Familiarización con la programación de aplicaciones para nodos IoT.
* Identificación del tipo de aplicaciones creadas y por crear.
* Creación de aplicaciones para los nodos IoT.
* Pruebas de las aplicaciones para los nodos IoT.
* Revisión y optimización de las aplicaciones.
* Configuración correcta de una red de malla LoRa.

En cuanto a la propuesta en sí, de momento, estamos evaluando qué opción es más factible e interesante. Según Felix (director del TFG) podemos hacer lo siguiente:

1. Deployment: del entorno local a "Internet". Fase 1: con VM, MQTT broker en Internet, las placas en casa. Fase 2: En Campus Nord (u otro lugar "real"), las placas en este lugar y conectado a miniPCs que podemos controlar remotamente (reproducir como tenemos el testbed).
2. Añadir sensores a las placas y extender que el monitoring envía valores de sensores (actualmente, el monitoring envía valores del LoRaMesher, o sea, de su propio software). Podríamos conectar sensores a las placas, leer los valores y enviarlos. Conlleva retoques al software, la carpeta de monitoring.
3. "Portar" los componentes de Jack al LoRaMesher/LoRaChat actual. Jack usó un LoRaMesher/LoRaChat de hace 1 año. Se podría integrar lo que ha hecho a la versión actual. ¿Qué supone? El trabajo de Jack es una carpeta, idealmente sería "copiar" la carpeta al nuevo LoRaMesher/LoRaChat actual, en la práctica, algunos calls habrán cambiado etc, así habría que hacer adaptarlo.
4. Validar con otras placas, no solo los T-Beams. Disponemos de otro tipo de placas con ESP32. Si funcionan, se podría validar y usar en el deployment para tener un deployment con 2 tipos de nodos.

En todos los casos, el trabajo se basa en mejorar el TFG de Jack (ya mencionado anteriormente).

## 2.2 Requisitos

A continuación, hay una lista de requisitos necesarios para el correcto desarrollo del proyecto, que se podrían clasificar según sean funcionales o no funcionales:

Funcionales:

* Adaptabilidad y compatibilidad con los demás nodos dentro de la red LoRa.
* Garantizar el uso correcto de las aplicaciones, lo que significa que no deben haber errores.
* Comentar y documentar el código para su fácil comprensión.

No funcionales:

* Los resultados deben ser de fácil acceso y servir de referencia a los que busquen mejorar este trabajo.
* Informarse del funcionamiento de redes LoRa y nodos IoT.
* Revisión continua del proyecto para asegurar el correcto avance.

## 2.3 Riesgos y obstáculos

Para el proyecto se han identificado los siguientes obstáculos y/o riegos que pueden afectar al desarrollo de este:

* Fecha límite: hay un plazo establecido para la entrega, por lo que se ha de gestionar bien el tiempo. Puede influir al tiempo que se dedique en cada tarea e impactar negativamente a la calidad del proyecto en caso de retraso.
* Falta de experiencia y disponibilidad de la información: no se ha trabajado antes con este tipo de aplicaciones, además de no haber una gran cantidad de información disponible. Es necesaria una fase de adaptación y comprensión de las tecnologías. Gran parte del código está desarrollado por otra persona, por lo que, hay que entender la base.
* Errores o retrasos: como ya se ha comentado, en caso de retraso, fallo de las aplicaciones o errores de programación afectan considerablemente la organización y planificación de tareas además de la calidad del trabajo.
* Fallos inesperados o incontrolables: en todo caso, aunque no sea muy probable, hay que tener en cuenta la posibilidad de una falla de hardware (tanto placas como el ordenador en cuestión), además de situaciones médicas imprevistas.
* Desconocimiento del final: en los proyectos de investigación, nos podemos encontrar que el camino a seguir no lleva a una solución exitosa. Lo que me recuerda a la siguiente frase “Caminante, no hay camino, se hace camino al andar”[[6]](#footnote-5), que sugiere que no hay un destino predefinido, sinó que, se construye a medida que avanzamos.

## 2.4 Metodología

Este proyecto adoptará una metodología Agile. Esta forma es la que más nos interesa debido a la transparencia y la comunicación, el desarrollo iterativo, la planificación adaptativa y la mejora continua. Se realizará una planificación con el fin de identificar cuándo se llevará a cabo cada cosa. Durante el desarrollo del proyecto, se realizarán reuniones con el director (usando Meet) para discutir los avances. En estas reuniones se podrán formar nuevas tareas y añadirlas al plan existente. De forma añadida, también habrá comunicación por correo electrónico con el director con el fin de resolver dudas.

# 3. Referencias

[1] *Projectes*. (n.d.). El Raco Fib. <https://raco.fib.upc.edu/projectes/el-meu-projecte>

[2] Salazar, Ó. (2023, 29 marzo). *¿Qué es la escala de madurez tecnológica (TRL)?* Euro Funding. Retrieved September 25, 2024, from <https://euro-funding.com/es/blog/que-es-la-escala-de-madurez-tecnologica-trl/>

[3] *Desenvolupament d’una infraestructura experimental per a xarxes LoRa en malla.* (n.d.). <https://raco.fib.upc.edu/projectes/veure-competencies-tfg.jsp?projecte=192799>

[4] *Haciendo IoT con LoRa: Capitulo 2.- Tipos y Clases de Nodos*. (2017, October 2). Medium. Retrieved September 25, 2024, from <https://medium.com/@Sabasacustico/haciendo-iot-con-lora-capitulo-2-tipos-y-clases-de-nodos-3856aba0e5be>

*[5] Griffiths, J. (2024, 06 27). Exploring Cross-network applications for a LoRa mesh network.*

*[6] Machado, A. (1998). Caminante. Planeta.*

1. *Projectes*. (n.d.). El Raco Fib. <https://raco.fib.upc.edu/projectes/el-meu-projecte> [↑](#footnote-ref-0)
2. Salazar, Ó. (2023, 29 marzo). *¿Qué es la escala de madurez tecnológica (TRL)?* Euro Funding. Retrieved September 25, 2024, from <https://euro-funding.com/es/blog/que-es-la-escala-de-madurez-tecnologica-trl/> [↑](#footnote-ref-1)
3. *Desenvolupament d’una infraestructura experimental per a xarxes LoRa en malla.* (n.d.). <https://raco.fib.upc.edu/projectes/veure-competencies-tfg.jsp?projecte=192799> [↑](#footnote-ref-2)
4. *Haciendo IoT con LoRa: Capitulo 2.- Tipos y Clases de Nodos*. (2017, October 2). Medium. Retrieved September 25, 2024, from <https://medium.com/@Sabasacustico/haciendo-iot-con-lora-capitulo-2-tipos-y-clases-de-nodos-3856aba0e5be> [↑](#footnote-ref-3)
5. *Griffiths, J. (2024, 06 27). Exploring Cross-network applications for a LoRa mesh network.* [↑](#footnote-ref-4)
6. *Machado, A. (1998). Caminante. Planeta.* [↑](#footnote-ref-5)