

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO EN
ESTANQUES PISCÍCOLAS BASADO EN EL INTERNET DE LAS COSAS (IOT)

MARÍA ALEJANDRA LOZANO JIMÉNEZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍAS
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES
BUCARAMANGA
2019

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO EN
ESTANQUES PISCÍCOLAS BASADO EN EL INTERNET DE LAS COSAS (IOT)

MARÍA ALEJANDRA LOZANO JIMÉNEZ

Trabajo de Grado para optar al título de
Ingeniera Electrónica

Director

Homero Ortega Boada

Ph.D. En Ciencias de la Ingeniería

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECAÑICAS
ESCUELA DE INGENIERÍAS
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES
BUCARAMANGA

2019

AGRADECIMIENTOS

Agradezco a mis padres Rosa y Heriberto por sus consejos, enseñanzas y constante apoyo para salir adelante. A mis hermanos John y José por impulsarme a ser mejor persona cada día.

A la Universidad Industrial de Santander y a los profesores de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones por la formación académica y personal recibida durante estos años.

A todos mis compañeros de carrera los cuales me acompañaron y apoyaron en este largo proceso de formación dejando recuerdos inolvidables.

Al profesor Homero Ortega Boada mi director de trabajo de grado, por dedicar su tiempo y experiencia para guiarme en el desarrollo del proyecto, además, por enseñarme a ver las cosas de una forma diferente.

CONTENIDO

	pág.
INTRODUCCIÓN	14
1. OBJETIVOS	18
1.1. OBJETIVO GENERAL	18
1.2. OBJETIVOS ESPECÍFICOS	18
1.3. ALCANCES	18
2. MODELO PARA UN SISTEMA PISCÍCOLA INTELIGENTE	20
2.1. REVISIÓN DEL ESTADO DEL ARTE	20
2.2. ECOSISTEMA PISCÍCOLA INTELIGENTE	22
2.3. MODELO PARA UN SISTEMA PISCÍCOLA INTELIGENTE	27
2.4. METODOLOGÍA PARA SOLUCIONES INTELIGENTES CON IOT	30
3. IDENTIFICACIÓN DE REQUISITOS Y DISEÑO FUNCIONAL	32
3.1. INGENIERÍA DE REQUERIMIENTOS	32
3.1.1. Perspectiva del producto	32
3.1.2. Funciones del producto	33
3.1.3. Características del usuarios	34
3.1.4. Definición de restricciones para la versión piloto	35
3.1.5. Requisitos específicos y casos de usos	35
3.2. PROCESO DE DISEÑO	42
3.2.1. Modelo general del sistema de monitoreo	42
3.2.2. Red del Terminal IoT	43
3.2.3. Diagrama de clases	46

4. CAPA IOT	49
4.1. NODO EN ESTANQUE	50
4.1.1. Sensores	50
4.1.2. Tarjeta de desarrollo	58
4.1.3. Fuente de alimentación	60
4.1.4. Descripción del funcionamiento	61
4.2. SUBCAPA DE TRANSMISIÓN LOCAL	62
4.2.1. Datos enviados	62
4.3. NODO CENTRAL	63
4.3.1. Módulo receptor	64
4.3.2. Raspberry Pi 3B+	64
4.3.3. Descripción del funcionamiento	65
4.4. SUBCAPA DE ACCESO	66
5. GATEWAY DE LA PLATAFORMA	68
5.1. API REST	68
5.2. BROKER MQTT MOSQUITO	71
6. CAPA DE LA NUBE	74
6.1. ALMACENAMIENTO EN LA BASE DE DATOS	74
6.2. EJEMPLO DE PROCESAMIENTO EN LA NUBE	75
6.3. CONTROL DE ACCESO A USUARIOS Y DISPOSITIVOS	76
7. CAPA CLIENTE	77
7.1. REGISTRO DE ESTANQUES	78
7.2. TOMA DE DATOS	78
7.3. MONITOR PISCICULTURA	79
8. VALIDACIÓN DE LA COMUNICACIÓN LOCAL	81

8.1. MONTAJE FINAL	81
8.2. PRUEBAS DE COMUNICACIÓN	82
8.2.1. Primera prueba	83
8.2.2. Segunda Prueba con mejoras en software	84
9. CONCLUSIONES	86
10. TRABAJO FUTURO	87
BIBLIOGRAFÍA	88

LISTA DE FIGURAS

	pág.
Figura 1. Ecosistema de piscicultura inteligente.	26
Figura 2. Modelo de capas para un sistema piscícola inteligente.	29
Figura 3. Diagrama de bloques del proyecto <i>Laboratorio para el aprendizaje de la piscicultura con sistema de producción automatizada y en ambiente virtual.</i>	33
Figura 4. Diagrama de bloques del Sistema de monitoreo en estanques piscícolas basado en IoT.	38
Figura 5. Diagrama de los Casos de uso para los actores del sistema.	42
Figura 6. Modelo general del sistema de monitoreo en estanques piscícolas basado en IoT.	43
Figura 7. Red del terminal IoT.	44
Figura 8. Diagrama de los clases para el software del sistema de monitoreo en estanques piscícolas basado en IoT.	48
Figura 9. Diagrama de bloques del terminal IoT.	49
Figura 10. Sensor DS18B20.	51
Figura 11. Conexión sensor DS18B20.	51
Figura 12. Sonda	53
Figura 13. Circuito pH EZO.	54
Figura 14. Sonda	56
Figura 15. Circuito OD EZO.	57
Figura 16. Placa portadora EZO aislada eléctricamente.	58
Figura 17. Pines tarjeta de desarrollo Lora32 de TTGO.	59

Figura 18.	Batería de litio 18650.	60
Figura 19.	Diagrama de funcionamiento del nodo en estanque.	62
Figura 20.	Raspberry Pi 3B+	64
Figura 21.	Diagrama de flujo de los dos hilos ejecutados en la Raspberry.	66
Figura 22.	Vista de la API creada para la variable oxígeno disuelto	70
Figura 23.	Diagrama de funcionamiento MQTT	72
Figura 24.	Modelo de la base de datos implementada en postgresSQL	75
Figura 25.	Sonda	80
Figura 26.	Modelo electrónico funcional final del nodo en estanque.	81
Figura 27.	Planos de los estanques piscícolas en la granja de la UIS <i>El Hangar</i> en Guatiguará. Nodos en estanques (puntos rojos), nodo central (punto verde).	82
Figura 28.	Nodo en estanque en la granja UIS <i>El Hangar</i> en Guatiguará.	83

LISTA DE TABLAS

	pág.
Tabla 1. Interfaces de comunicación del sistema de monitoreo.	37
Tabla 2. Requisitos de software para el sistema de monitoreo.	37
Tabla 3. Caso de uso: Almacenar la información.	40
Tabla 4. Caso de uso: Visualizar variables.	40
Tabla 5. Caso de uso: Controlar toma de datos.	41
Tabla 6. Caso de uso: Registrar estanque.	41
Tabla 7. Comparación de tecnologías de comunicación.	45
Tabla 8. Sensores de temperatura en líquidos.	50
Tabla 9. opciones para medir pH en líquidos.	52
Tabla 10. Opciones para medir oxígeno disuelto en líquidos.	55
Tabla 11. Tarjetas de desarrollo	59
Tabla 12. Características Raspberry Pi 3B+.	65
Tabla 13. Resultados prueba enlace LoRa	84
Tabla 14. Resultados prueba 2 enlace LoRa con mejoras en software.	85

RESUMEN

TÍTULO: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO EN ESTANQUES PISCÍCOLAS BASADO EN EL INTERNET DE LAS COSAS (IOT) *

AUTOR: MARIA ALEJANDRA LOZANO JIMÉNEZ **

PALABRAS CLAVE: PISCICULTURA, INTERNET DE LAS COSAS, ECOSISTEMA PISCÍCOLA INTELIGENTE, MONITOREO, SENSORES.

DESCRIPCIÓN:

Se propone un modelo para un sistema piscícola inteligente y se realiza la implementación de una primera versión de un servicio basado en el Internet de las Cosas (IoT) de apoyo a los piscicultores. El servicio se orienta al monitoreo de las variables del agua y su implementación abarca diversas técnicas convergentes: Los Terminales IoT que comprenden sensores, actuadores, una red Lora para la comunicación local, todo esto organizado en nodos que se alojan en los estanques y un nodo central que permite el acceso a Internet. Las soluciones en la nube incluyen almacenamiento centralizado, control de acceso de usuarios y dispositivos así como un pequeño módulo de procesamiento de datos para mostrar la viabilidad de usar todas estas posibilidades. La aplicación web permite a los usuarios remotos controlar la toma de datos del sistema y acceder a la información de las variables del agua de los estanque desde cualquier dispositivo con acceso a internet.

Una de las mayores cualidades del desarrollo presentado es la escalabilidad para poder conectar tantos elementos de hardware y software como sea necesario a nivel de acceso como en la nube, incluyendo técnicas de machine learning. El modelo IoT propuesto, se apoya en los estándares internacionales, pero incluye novedades al nivel de detalle en cada capa para que los desarrolladores de cualquier otro tipo de soluciones IoT tengan mayores facilidades a la hora de seleccionar las tecnologías IoT más apropiadas, entre la amplia gama disponible, para una solución particular.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y telecomunicaciones. Director: Homero Ortega Boada, Ph.D. En Ciencias de la Ingeniería.

ABSTRACT

TITLE: DESIGN AND IMPLEMENTATION OF A SYSTEM OF MONITORING IN FISH TANKS BASED ON THE INTERNET OF THINGS (IOT) *

AUTHOR: MARÍA ALEJANDRALOZANO JIMÉNEZ **

KEYWORDS: PISCICULTURE, INTERNET OF THINGS, SMART FISH FARM ECOSYSTEM, MONITORING, SENSORS.

DESCRIPTION:

A model for an intelligent fish farming system is proposed and the implementation of a first version of a service based on the Internet of Things (IoT) to support fish farmers is carried out. The service is aimed at monitoring water variables and its implementation cover various convergent techniques: IoT terminals that include sensors, actuators, a Lora network for local communication, all organized in nodes that are housed in the tanks and a node central that allows access to the Internet. Cloud solutions include centralized storage, user and device access control as well as a small data processing module to show the feasibility of using all these possibilities. The web application allows remote users to control the data collection of the system and access the information of the water tank variables from any device with internet access.

One of the greatest qualities of the development presented is the scalability to connect as many hardware and software elements as necessary at the access level as in the cloud, including machine learning techniques. The proposed IoT model is based on international standards, but includes developments at the level of detail in each layer so that developers of any other type of IoT solutions have greater facilities when selecting the most appropriate IoT technologies, among the wide range available, for a particular solution.

* Bachelor Thesis

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y telecomunicaciones. Director: Homero Ortega Boada, Ph.D. En Ciencias de la Ingeniería.

INTRODUCCIÓN

Imaginemos un criadero donde los peces se desarrollan en unas condiciones que están limitadas en espacio pero al mismo tiempo les permiten crecer en un tiempo óptimo debido al confort que les brinda un conjunto de tecnologías y métodos. Eso, sumado a estrategias apropiadas de negocios es lo que podríamos llamar un ecosistema piscícola inteligente. Precisamente los grupos Simon, Radiogis, el Ipred y el CentroTIC se han unido para desarrollar el proyecto *“Laboratorio para el aprendizaje de la piscicultura con sistema de producción automatizada y en ambiente virtual”*, en adelante LAPIS, que busca dar los primeros pasos en esa dirección. El Internet de las cosas (IoT) juega un papel muy importante ya que los diferentes dispositivos usados en los estanques pueden resultar siendo parte de la Internet uniéndose a recursos allí presentes para satisfacer necesidades tanto de los peces como de los usuarios remotos.

Con los sistemas cerrados de producción piscícola se logran varios beneficios entre ellos, la disminución de enfermedades al no estar tan expuestos a todos los factores ambientales ¹, mayor producción y productividad debido a que se pueden tener más peces por unidad de superficie y además, se tiene un mayor control de la calidad del agua en la que viven los peces. Un problema que estos sistemas afrontan es que se desperdicia mucho alimento y además ese desperdicio eleva la contaminación del estanque. El suministro de alimento se realiza teniendo en cuenta la especie de peces, el peso y el rango de temperatura del agua. Muchos proveedores de alimento proporcionan tablas que indican el alimento necesario pero hay muchos factores

¹ FAO. *Visión general del sector acuícola nacional Colombia*.

que no son tenidos en cuenta, razón por la cual las raciones distan de ser precisas y puede presentarse exceso o falta de alimento en los peces.

En una primera fase el proyecto solo busca atender las necesidades básicas de los usuarios investigadores que tienen la tarea de realizar diferentes estudios para poder visionar un ecosistema piscícola inteligente que resulte viable para las condiciones de Santander. En futuras fases se buscará atender las necesidades de nuevos actores externos a la universidad como pequeños, medianos y grandes piscicultores y las demás entidades que pueden entrar en el juego de la producción y los negocios de la piscicultura. Pero el proyecto LAPIS limita aún más sus alcances para priorizar los estudios que permitan relacionar las variables del estanque con la cantidad óptima de alimento y de condiciones de confort que permita una óptima velocidad en el crecimiento de los peces al mismo tiempo que se disminuyen costos por contaminación del estanque, entre otras cosas.

Por otro lado, el CentroTIC, en el marco del proyecto *“Plataforma IoT para el desarrollo de servicios inteligentes de monitoreo ambiental”*, ha venido implementando una plataforma escalable para soportar los más diversos servicios que se desarrollan en la UIS. La piscicultura inteligente es uno de ellos. Busca atender todas las necesidades técnicas de la piscicultura como el sensado de información, almacenamiento de los datos en la nube, procesamiento de esos datos en la nube, uso de técnicas de inteligencia artificial y aplicaciones que atiendan las necesidades de los usuarios.

En medio de los estudios que realiza la UIS en piscicultura se destaca el uso de estanques de geomembrana que permiten elevar sustancialmente la densidad de peces en un estanque. En este sentido, poder contar con mediciones de la calidad del agua que sean acertadas, permanentes y de bajo costo para luego relacionarlas

con la alimentación es una necesidad bastante importante. Por esa razón, el presente proyecto ha sido desarrollado para atender la necesidad de monitorear electrónicamente y de manera permanente las variables del agua que más influyen en el confort de los peces usando las técnicas modernas del Internet de las cosas (IoT).

De manera más precisa, el objetivo general del proyecto es el de diseñar e implementar un sistema de monitoreo en estanques piscícolas basado en el internet de las cosas (IoT) que sea en tiempo real, con fácil acceso a la información y escalable para aplicar nuevas técnicas como las relacionadas con ciencia de datos.

En los siguientes capítulos se documenta el desarrollo del proyecto explicando de manera sencilla los elementos más relevantes en cada etapa. Toda esta información se organiza de la siguiente manera: en el capítulo 1 se muestran los objetivos del proyecto; en el capítulo 2 se realiza una revisión general del estado del arte, luego se propone un modelo para un sistema piscícola inteligente. En el capítulo 3 se identifican los requisitos y se realiza el diseño de la solución planteando el modelo de capas particularmente para el sistema de monitoreo, posteriormente se elabora el diagrama de clases para el software a implementar. Los siguientes capítulos documentan el desarrollo de la solución en cada capa presentada en el modelo diseñado en el capítulo 3. El capítulo 4 corresponde a la Capa IoT, donde se describe la implementación del terminal IoT; en el capítulo 5 se habla de la Capa de componentes en la nube, donde se presentan las herramientas usadas para la gestión de dispositivos y luego se explican los servicios como: el almacenamiento de la información, un ejemplo de procesamiento en la nube y el control de acceso a usuarios y dispositivos; el capítulo 6 aborda la capa cliente, donde se expone de manera general la implementación de la aplicación web. Finalmente, en el capítulo 7 se presenta la validación experimental de la comunicación del sistema en un estanque piscícola de

la granja El Hangar en Guatiguará, por medio de pruebas en la comunicación local del sistema donde se busca comprobar que la información recibida por nodo central, corresponda con la enviada desde el nodo en estanque.

Con la finalización de este proyecto se ha logrado automatizar la toma de datos de las variables del agua que tienen más influencia en el crecimiento de los peces permitiendo el acceso de los usuarios remotos no solo para consultar información a través de una aplicación web, sino también permitiéndole controlar la toma de datos. Gracias a este desarrollo es posible contar con un registro completo, durante todo el día, del comportamiento del agua en un estanque sin la necesidad de tener un operario dedicado a esta tarea, mientras, al mismo tiempo, se logra elevar la confiabilidad de las mediciones. De igual manera, aunque este trabajo sea sólo la semilla de la piscicultura inteligente, cabe resaltar que cuando estos sistemas se popularicen, tendrán impactos a largo plazo en Santander. Sí, en el plano económico se espera una reducción de costos de producción en esta actividad; en el plano social se espera elevar el número de piscicultores en la región; en el investigación, se habrán creado las condiciones para hacer investigación, desarrollo e innovación, incluyendo técnicas de inteligencia artificial para elevar permanentemente la producción piscícola.

1. OBJETIVOS

1.1. OBJETIVO GENERAL

- Diseñar e implementar un sistema de monitoreo en estanques piscícolas basado en el internet de las cosas (IoT).

1.2. OBJETIVOS ESPECÍFICOS

- Monitorear la temperatura, el pH y oxígeno disuelto en el agua de un estanque piscícola con un modelo electrónico funcional del terminal IoT.
- Visualizar los datos monitoreados por medio de una aplicación web implementada en una plataforma IoT.
- Validar experimentalmente la comunicación del sistema en un estanque piscícola de la granja *El Hangar* en Guatiguará.

1.3. ALCANCES

- Se definirán los requerimientos del sistema con base en las necesidades del equipo de trabajo y de los usuarios.
- El diseño general del sistema busca definir el modelo de arquitectura a implementar y el diseño de la red del terminal IoT, con el fin de cumplir con los requerimientos establecidos.
- La adquisición de los datos, se limita a leer las señales de los sensores de temperatura, pH y oxígeno disuelto.

- El terminal IoT estará conformado por: sensores para medir temperatura, oxígeno disuelto y pH en el agua, una Raspberry, una tarjeta de desarrollo con módulo de transmisión para la recolección de datos y el envío.
- La aplicación web será desarrollada en la plataforma IoT del centro TIC y permitirá la visualización de las mediciones realizadas en el agua del estanque piscícola, además, será accesible a los usuarios desde cualquier dispositivo con conexión a internet.
- Se validará experimentalmente la comunicación mediante pruebas que buscan comprobar que la información recibida en el nodo central, corresponda con la enviada desde el nodo en estanque. Estas pruebas se realizarán en un estanque piscícola de la granja UIS *El Hangar* en Guatiguará.

2. MODELO PARA UN SISTEMA PISCÍCOLA INTELIGENTE

En este primer capítulo se documentan los aspectos más relevantes encontrados luego de realizar una revisión del estado del arte de los sistemas piscícolas en el mundo. Luego se plantea un ecosistema piscícola inteligente y finalmente se propone un modelo de capas para la implementación de servicios basados en IoT de apoyo a la piscicultura inteligente.

2.1. REVISIÓN DEL ESTADO DEL ARTE

Para plantear un modelo completo para la piscicultura inteligente primero se realizó una revisión bibliográfica de los trabajos realizados en esta área para tomar los elementos buenos de cada desarrollo y además, analizar que componentes faltan en las implementaciones actuales pero que de igual manera son importantes tenerlos en cuenta. A continuación se resumen los más relevantes:

En la Universidad de Agricultura de China ubicada en Beijín, implementaron un sistema de monitoreo de la calidad del agua en línea para cultivos intensivos de peces en China, que combina tecnologías como un servidor web y telecomunicaciones móviles. Además, basado en datos históricos, este sistema pronostica la calidad del agua con redes neuronales artificiales (ANN) y permite controlar la calidad del agua para reducir las pérdidas catastróficas ².

En la Universidad del Quindío en Colombia se construyó un sistema de instrumenta-

² X. ZHU y col. "A remote wireless system for water quality online monitoring in intensive fish culture". En: *El Sevier Computers and Electronics in Agriculture* (2010).

ción que mide temperatura, pH y oxígeno disuelto. El sistema cuenta con una etapa de control manejada por el microcontrolador ATmega8 que se encarga de la adquisición de las variables, además, cuenta con una aplicación desarrollada en C++ Builder para generar reportes de las mediciones realizadas en archivos de texto en formato .txt para realizar de forma manual el análisis de dicha información ³.

Los investigadores de la Universidad Can Tho en Vietnam por medio de una red ZigBee de sensores inalámbricos monitorean las condiciones ambientales críticas, el control se realiza con ayuda de microcontroladores MSP430 de baja potencia de Texas Instruments. Este sistema recopila, analiza y presenta los datos en una interfaz gráfica de usuario en LabVIEW. Además, mediante un servicio de pasarela SMS, el sistema envía mensajes de alerta con prontitud al usuario y permite la activación de la bomba de oxígeno también con un mensaje del número de teléfono móvil autorizado ⁴.

Como conclusión de la revisión bibliográfica se pudo encontrar que China es uno de los países líderes implementando sistemas de producción piscícolas de la mano de la tecnología, además, en el trabajo mencionado llevado a cabo en Beijín aplican algoritmos de procesamiento, lo cual es bastante interesante y puede tener diversas aplicaciones en el campo de la piscicultura. Por otro lado, se encontró que gran parte de los sistemas de monitoreo implementados sólo permiten el acceso a la información a nivel local y pocos llevan la información a internet, por consiguiente

³ A. A. NAVARRO, J. B. PADILLA y J. J. PRÍAS. "Construcción de un Sistema de Instrumentación para la Medición de la Temperatura, pH y Oxígeno Disuelto presentes en la Piscicultura bajo Condiciones de Estanque Artificial". En: *Scientia et Technica Año XVIII* 18 (2013).

⁴ N. T. K. DUY y col. "Automated monitoring and control system for shrimp farms based on embedded system and wireless sensor network". En: *In Electrical, Computer and Communication Technologies (ICECCT), 2015 IEEE International Conference on* (pp. 1-5) 18 (2013), págs. 1-5.

sería una característica útil integrar el internet de las cosas a estos sistemas para poder acceder a la información y controlar el sistema desde cualquier lugar con internet, además, para crear un respaldo de la información en la nube.

2.2. ECOSISTEMA PISCÍCOLA INTELIGENTE

En Colombia la acuicultura ⁵ ha tenido un crecimiento promedio del 13 % anual en los últimos 27 años, destacándose los pequeños y medianos acuicultores. Además, esta actividad ha ido reemplazando la producción pesquera nacional de captura, la cual se ha visto afectada por los cambios climáticos. Dicha situación ha posicionado al país en la sexta posición de producción acuícola en América Latina ⁶.

Sin embargo, aunque este promedio de crecimiento supere con mucha diferencia a otras actividades agropecuarias, la acuicultura ha surgido de manera desordenada, sin planificación y sin una política gubernamental que sirva de apoyo al impulso de la actividad. Por tal razón, existe una oportunidad muy grande en el desarrollo de esta actividad, en especial en la piscicultura que contempla en específico la cría de peces. Mediante una implementación organizada enfocada en solucionar las principales necesidades con ayuda de las tecnologías, se puede aumentar significativamente la rentabilidad de este negocio.

⁵ C. ENCINAS y col. "Design and implementation of a distributed IoT system for the monitoring of water quality in aquaculture". En: *Conference: 2017 Wireless Telecommunications Symposium (WTS)* (2017).

⁶ Sara Patricia y BAGES Fernando MERINO María Claudia y BONILLA. *Diagnóstico del estado de la Acuicultura en Colombia*. Plan Nacional de Desarrollo de la Acuicultura Sostenible en Colombia. Colombia: Ministerio de Agricultura y Desarrollo Rural, Autoridad Nacional de Pesca y Acuicultura (AUNAP), Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO), 2013.

Actualmente, la piscicultura se está implementando cada vez más en tanques de geomembrana, el cual, es un material derivado del petróleo y altamente impermeable. Esta nueva tendencia, permite el aprovechamiento de suelos no aptos para la acuicultura tradicional, facilidad para implementar procesos de recirculación y mayor optimización del uso del espacio. Por ejemplo, mientras que en los sistemas tradicionales siembran entre 1 y 5 peces por metro cuadrado, si se cosechan de una libra, se obtendrían de 2 a 3 kilos por metro cúbico; mientras que en los tanques de geomembrana, con un diseño técnico adecuado y unos equipos apropiados puede lograrse hasta 100 kilos por metro cubico ⁷.

Por otro lado, el factor de mayor de impacto en la rentabilidad y productividad del negocio es la cantidad de alimento suministrado, que depende de la temperatura del agua (generalmente con temperaturas bajas, los peces no salen a comer) lo que influye principalmente en el peso del animal. Esta última situación, ha generado que se realicen mediciones periódicas del peso, sacando del tanque una muestra de peces lo cual les genera gran estrés y retrasan su crecimiento. Cuando se les suministra alimento de más, el alimento se desperdicia y contamina el estanque, cuando se suministra menos de lo suficiente, se retrasa el crecimiento de los peces.

Garantizar las condiciones más adecuadas para el crecimiento de los peces y una producción altamente rentable, es posible en un ecosistema de piscicultura inteligente como el mostrado en la Fig. 1 donde se contemplan todas las etapas del ciclo de vida de la producción piscícola. La primera etapa es el acondicionamiento del medio, en este caso de los estanques, donde es necesario realizar ciertos tratamientos de desinfección del medio, para evitar enfermedades en los peces en un

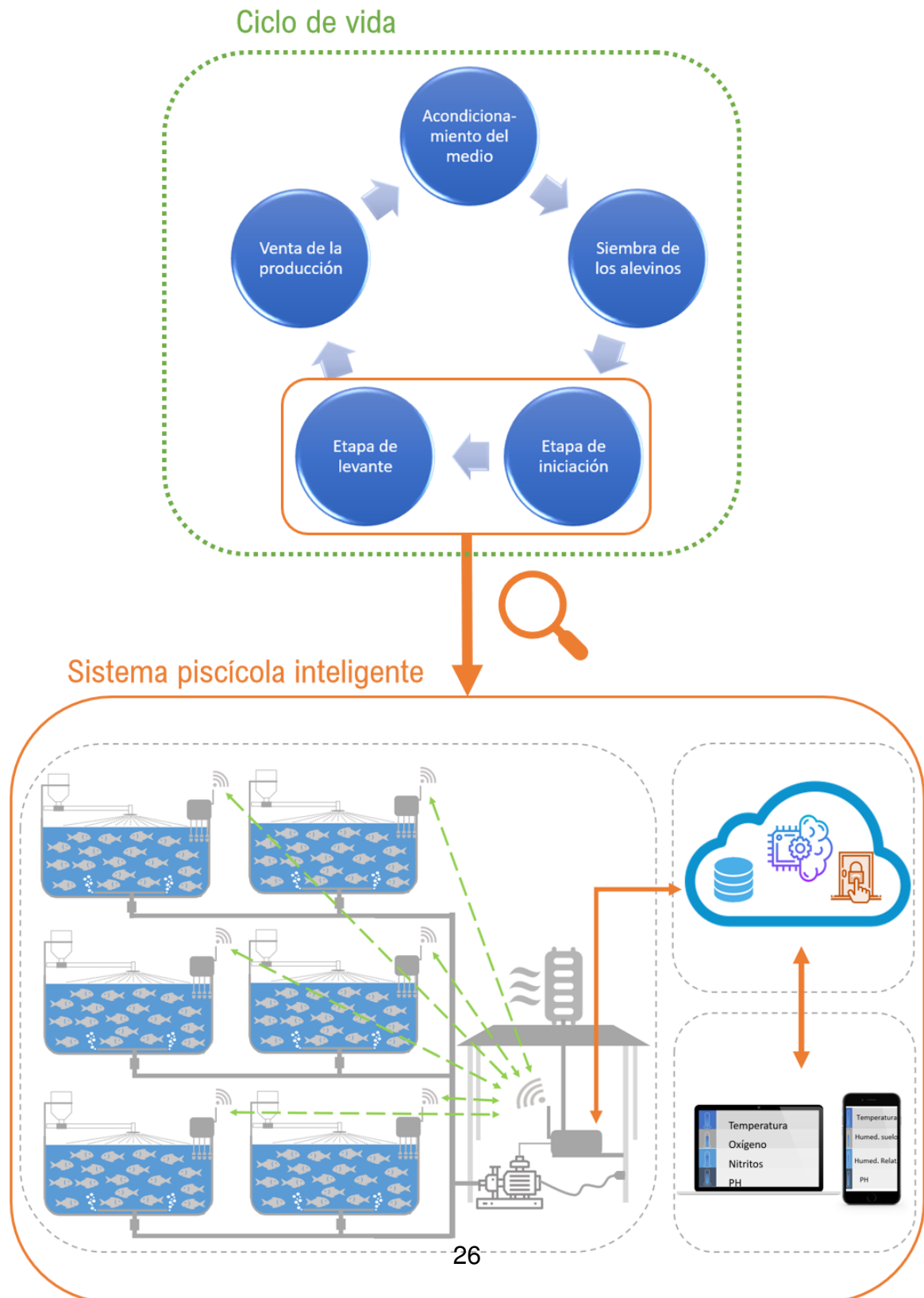
⁷ AUNAP. *La geomembrana, una opción para la piscicultura*. 2016.

futuro. La siembra es la segunda etapa, para lo cual implica la compra de los alevinos a un distribuidor, esto se costea con una inversión inicial o con el dinero de la producción anterior; posteriormente se procede al llenado del estanque y se ingresan los alevinos. La tercera y cuarta etapa corresponde a la iniciación y el levante, estas dos etapas contemplan el crecimiento de los peces y se lleva a cabo en los estanques. En esta etapa, los actores son el investigador y el productor, quienes son los que interactúan con el sistema inteligente piscícola inteligente que consiste en un sistema cerrado de crianza en tanques de geomembrana donde la mayor parte de actividades son automatizadas. Este sistema monitorea el estado de cada uno de los estanques, luego, esta información se transmite a un nodo central, el cual es el encargado de enviarla a la plataforma IoT en la nube. Allí, la información se almacena en una base de datos y además, se analiza y procesa mediante algoritmos de ciencia de datos, los cuales determinan el envío de acciones al sistema local (estanques) como por ejemplo la cantidad de alimento que los alimentadores automáticos deben suministrar y cuando es necesario activar la bomba o la calefacción. También, toda la información del monitoreo recibida y el resultado del procesamiento en la nube, se pone al alcance del usuario por medio de una aplicación web, donde se puede visualizar en tiempo real el estado de sus estanque y además controlar manualmente los actuadores.

Este sistema piscícola inteligente permite minimizar la intervención humana en el proceso de producción para no generar estrés en los animales. Finalmente, la última etapa del ciclo es la venta de la producción, donde el productor interactúa con el cliente o directamente con el consumidor para vender su producción y obtener dinero. Esta actividad le permite obtener las ganancias del negocio y adquirir recursos para invertir en la nueva producción. Un ecosistema de piscicultura inteligente cuenta principalmente con las siguientes características:

- Limpieza y tratamientos para el acondicionamiento del medio antes de la cosecha. También se aplican buenas prácticas para el uso eficiente del agua y manejo apropiado de los sedimentos .
- Estudio constante de la eficacia del alimento y de la velocidad de crecimiento de los peces. Además, se realiza una planificación adecuada de cada una de las etapas del ciclo de vida de producción.
- Registro de todas las inversiones y ganancias del sistema de producción para estimar costos y poder conocer la rentabilidad de la producción. Además, con todos los datos registrados se realiza una predicción de ganancias con ayuda de las herramientas de análisis de datos.
- Peces tranquilos con condiciones óptimas que genera su rápido crecimiento.
- Monitoreo electrónico constante de todas las variables que influyen en la producción.
- Ciencia de datos implementada para determinar la cantidad exacta de alimento requerida por los peces según las condiciones actuales del estanque.
- Sistema de alarmas ante valores críticos y sistema de control automatizado con un alimentador automático que suministre la cantidad necesaria de alimento, activación de la bomba cuando los niveles de oxígeno disuelto caigan, calefacción y ventilación para mantener la temperatura un punto óptimo para el crecimiento de los peces.
- Disponibilidad de la información en tiempo real y control por medio de una aplicación web accesible desde cualquier lugar accediendo mediante un dispositivo con conexión a internet. Además, también se tiene acceso a una aplicación local en tiempo real desde el sitio de producción sin necesidad de contar con internet.

Figura 1. Ecosistema de piscicultura inteligente.



2.3. MODELO PARA UN SISTEMA PISCÍCOLA INTELIGENTE

Para poder desarrollar un sistema piscícola inteligente es importante realizar una planeación, mediante un modelo que nos permita identificar los componentes y tecnologías requeridos en todos los niveles de la solución. Para este fin se propone el modelo de capas mostrado el figura 2 el cual se basa en el Modelo de referencia de IoT planteado por la Unión Internacional de Telecomunicaciones (UIT) ⁸, pero se le ha incluido mayor detalle para identificar diversas tecnologías disponibles. El modelo presenta los principales componentes del sistema piscícola inteligente organizados en tres capas y un gateway de la plataforma. A continuación se explican los principales aspectos de cada uno de estos elementos:

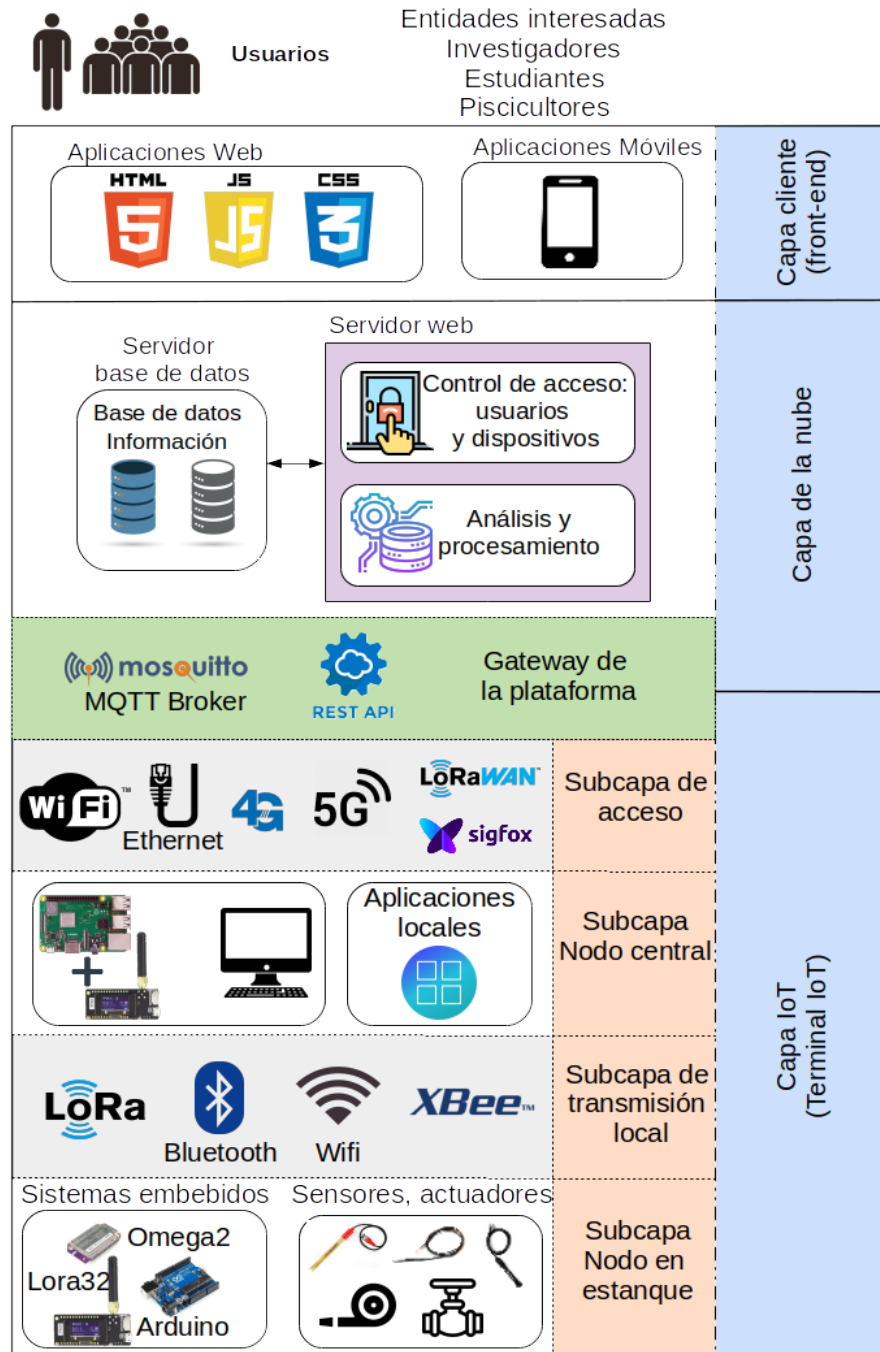
1. **Capa IoT:** Esta capa contiene a los terminales IoT, que hace referencia a los dispositivos con la capacidad de conectarse a internet para el envío y/o recepción de información. El terminal IoT puede ser tan sencillo o tan complejo como queramos, esto depende de las necesidades específicas del lugar donde se vaya a implementar. Sin embargo en este modelo se muestra la versión más compleja que consta de 4 subcapas: la subcapa del nodo en estanque el cual se encarga del monitoreo y control en cada estanque, por medio de sensores y actuadores que son manejados mediante un sistema embebido; la subcapa de transmisión local se encarga de transmitir la información de todos los estanques a nivel local hasta un punto central; la subcapa del nodo central se encarga de centralizar la información de los estanques, además, puede contar con aplicaciones a nivel local; y finalmente tenemos la subcapa de acceso la cual le permite al nodo central conectarse a internet por medio de diferentes tecnologías de comunicación como Wifi, Ethernet, 3G, 4G, Sigfox,

⁸ UIT. *Descripción general de Internet de los objetos*. 2012.

LoRaWAN entre otras.

2. **Gateway de la plataforma:** Este elemento hace referencia a los recursos utilizados para la gestión de los dispositivos que permiten que se mantengan conectados y comunicados con el servidor principal. Por lo tanto, existen componentes de estos recursos tanto en la capa IoT como en la capa de la nube, los cuales permiten el intercambio de información en ambos sentidos.
3. **Capa de la nube:** En esta capa se realiza el almacenamiento de la información en bases de datos, el control de acceso de usuarios y dispositivos, y el análisis y procesamiento de la información. Todos los procesos que se realizan en esta capa hacen parte del back-end analizándolo desde el punto de vista de un servicio, porque corresponde a todo el trabajo que se realiza para ofrecer el servicio pero es algo que está oculto para los usuarios.
4. **Capa de aplicaciones:** Esta capa hace referencia a los elementos con los que los usuarios finales interactúan o acceden a un servicio, constituye lo que se conoce como frontend de una solución. Como por ejemplo las aplicaciones web y las aplicaciones móviles.

Figura 2. Modelo de capas para un sistema piscícola inteligente.



2.4. METODOLOGÍA PARA SOLUCIONES INTELIGENTES CON IOT

Teniendo en cuenta el modelo de la Fig. 2 que se planteo en la sección anterior, también se hace necesario plantear una metodología que nos permita desarrollar e implementar las soluciones inteligentes con IoT basadas en este modelo. A continuación se muestran la metodología que se siguió para el desarrollo de este proyecto. Esta consiste básicamente en primero consultar acerca de los antecedentes existentes de la solución a implementar, luego, se definen los requerimientos del sistema y se realiza un diseño, finalmente se procede a la implementación del sistema capa por capa. Además, se explica en que consiste la aplicación de cada paso de esta metodología en el caso específico para el sistema de monitoreo en estanques piscícolas basado en el internet de las cosas (IoT), el cual es el objetivo de este proyecto.

1. **Revisión bibliográfica:** Revisión del estado del arte relacionado con sistemas de monitoreo piscícolas y de los antecedentes del proyecto *Laboratorio para el aprendizaje de la piscicultura con sistema de producción automatizada y en ambiente virtual*.
2. **Identificación de requerimientos:** En este paso de la metodología, se procede a identificar los requisitos funcionales y no funcionales del sistema según las necesidades de los usuarios. Además, se identifican los requisitos de las interfaces del usuario, software, hardware y comunicaciones.
3. **Diseño del sistema:** Se particulariza a esta solución el modelo de capas de la Fig. 2. Además, se realiza un diseño general de la red de comunicaciones local y del software.
4. **Implementación de la capa IoT:** Se realiza la adquisición de las señales de los sensores con ayuda de un sistema embebido, luego se integran todos los

componentes necesarios para el modelo electrónico funcional del terminal IoT, y finalmente se transmite la información a la plataforma IoT.

5. **Implementación de la capa nube:** Creación y configuración de la base de datos en la plataforma IoT, control acceso de usuarios y dispositivos.
6. **Implementación de la capa cliente:** Elaboración de la interfaz gráfica para la visualización de datos.
7. **Validación:** Se realizarán pruebas de las comunicaciones de la red interna del terminal IoT en un estanque piscícola, para verificar que lo que se transmite corresponde con lo que se recibe.

3. IDENTIFICACIÓN DE REQUISITOS Y DISEÑO FUNCIONAL

En este capítulo se documenta la ingeniería de requerimientos que parte de las necesidades de los usuarios para plantear los caso de usos del sistema, luego se define el modelo arquitectural de esta solución IoT y finalmente se procede al diseño del software del sistema por medio de un diagrama de clases, buscando siempre cumplir con los requerimientos planteados. Además, este capítulo corresponde al desarrollo de la primera etapa de la metodología de desarrollo de un solución IoT.

3.1. INGENIERÍA DE REQUERIMIENTOS

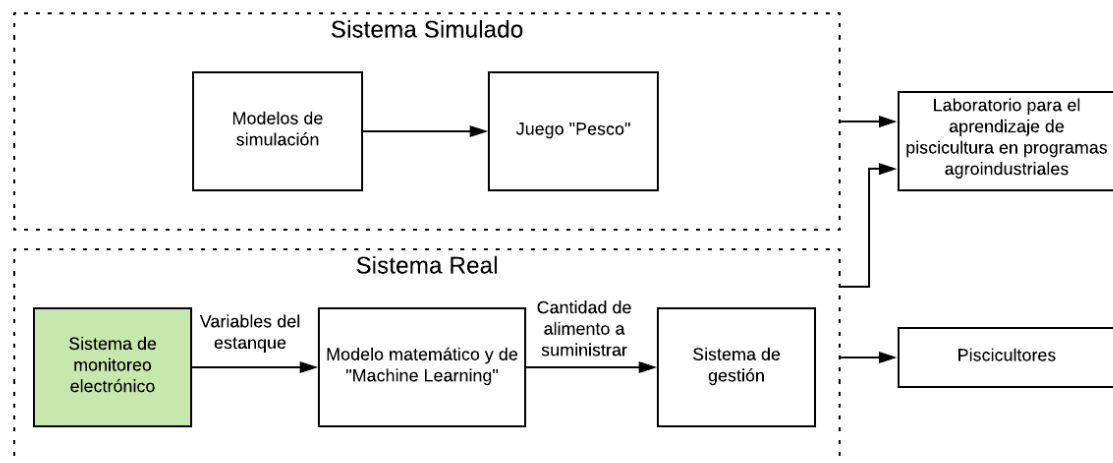
Para el diseño del sistema de monitoreo de un estanque piscícola se identifican los requerimientos que se basan principalmente en las necesidades de los usuarios del servicio, en este caso se centró la atención de usuarios que son investigadores, como es el caso de los profesores del IPRED Carlos Aníbal y Helman Eduardo Cabra, quienes son los que llevan el control de los estanques piscícolas de la granja de la UIS *El Hangar*. Para la identificación de requisitos de este proyecto se decide basarse en el estándar IEEE STD 830-1998 ⁹, el cual contempla los aspectos más importantes a tener en cuenta en la ingeniería de requerimientos de software, pero en este caso, lo aplicamos para para el hardware y software que involucra una solución IoT.

3.1.1. Perspectiva del producto El sistema de monitoreo en estanques piscícolas basado en el internet de las cosas, es un componente de un sistema de producción piscícola en desarrollo, con soporte informático para la gestión orientada a

⁹ IEEE. *IEEE Std 830-1998- IEEE Recommended Practice for Software Requirements Specifications*. 2009.

minimizar los pesajes y la pérdida de alimento. El proyecto general, contempla un sistema simulado y uno real como se muestra en la Fig. 3, para constituir un laboratorio para el aprendizaje de piscicultura. El principal papel del sistema de monitoreo electrónico dentro del proyecto general, es la adquisición y transmisión de las variables de los estanques para que posteriormente, esta información sea procesada por un modelo matemático y de *Machine Learning*. Luego del procesamiento se busca conocer la cantidad exacta de alimento que se le debe suministrar a los peces sin que haya desperdicio y sin retrasar el crecimiento de los peces.

Figura 3. Diagrama de bloques del proyecto *Laboratorio para el aprendizaje de la piscicultura con sistema de producción automatizada y en ambiente virtual*.



3.1.2. Funciones del producto El sistema de monitoreo en estanques piscícolas basado en IoT, tendrá 3 funciones principales:

- Adquisición de las variables del estanque.
- La transmisión de la información a la plataforma IoT.
- Visualización de la información en una aplicación web accesible por internet.

3.1.3. Características del usuarios Los servicios que ofrece el sistema de monitoreo para estanques piscícolas, podrán ser utilizados por personal académico para la investigación, estudiantes de programas agroindustriales del IPRED, piscicultores, entre otros. Sin embargo, para el levantamiento de requerimientos sólo se tendrán en cuenta los principales actores interesados actualmente, quienes son: 2 profesores del IPRED, y los grupos de Investigación Radiogis y Simon. A continuación se presentan las necesidades manifestadas por los principales actores del sistema:

Grupo SIMON y profesores del IPRED

- Sensado automático de temperatura, ph y oxígeno disuelto en al agua del estanque.
- Seguridad de la información ante pérdidas en las transmisiones.
- Almacenamiento centralizado de los variables.
- Diseño de un sistema de monitoreo escalable para incluir más estanques.
- Arquitectura del sistema que permita en un futuro adicionar más sensores e implementar el control del sistema.
- El nodo sensor debe estar protegido por una carcasa que soporte la intemperie.

Grupo RADIOGIS

- Solución IoT escalable a nivel de software y hardware.
- Diseño del sistema teniendo en cuenta el modelo planteado para soluciones IoT en Radiogis.
- El sistema debe ser inalámbrico y funcionar con baterías recargables.

- Se debe poder conectar y desconectar los sensores de manera ágil.
- Solución que demuestre la comunicación bidireccional con la plataforma del Centro TIC.
- Aplicación web accesible desde cualquier dispositivo con conexión a internet.
- La aplicación web debe permitir visualizar los datos en línea.
- Opción para descargar los datos registrados.

3.1.4. Definición de restricciones para la versión piloto

- El costo del hardware usado para el sistema de monitoreo no debe pasar los 5.000.000 millones de pesos colombianos destinados para ese fin en el proyecto.
- Los sensores para medir las variables del agua deben ser aptos para estar sumergidos por largos periodos de tiempo.
- Para los propósitos del presente trabajo, no se implementaran actuadores, pero la infraestructura hardware y software del sistema permite incorporarlos fácilmente.
- El cliente y el servidor se comunicaran mediante HTTP, el servidor con los terminales IoT mediante el protocolo MQTT y los terminales IoT con el servidor por HTTP, ya que así está definido en la plataforma IoT del Centro TIC.

3.1.5. Requisitos específicos y casos de usos

3.1.5.1. Requisitos de las interfaces

- **Interfaces del usuario:** La aplicación web es el componente del sistema de monitoreo con el cual el cliente interactuará principalmente, por tanto tendrá las siguientes opciones:

1. Registrar estanque: Esta área permite registrar un nuevo estanque en el sistema, para esta acción se requiere inicio de sesión con el usuario administrador del sistema piscícola.
2. Toma de datos: En esta sección el usuario administrador deberá iniciar sesión y posteriormente podrá ACTIVAR/DESACTIVAR la toma de datos.
3. Monitor: Seleccionar el estanque de interés, visualizar los datos actualizados de las variables del estanque seleccionado y sus respectivas gráficas y descarga de datos registrados.

- **Componentes hardware**

1. Sondas y sensores de pH y oxígeno disuelto de las marca Atlas Scientific con sus respectivas sustancias calibradoras.
2. Tarjetas aisladoras eléctricamente.
3. Sonda sensor de temperatura para agua.
4. Sensores de temperatura y humedad para poner dentro de la caja del nodo en estanque.
5. Tarjetas de desarrollo *LoRa32* de TTGO con módulo microcontrolador ESP32 y módulo de comunicación LoRa con antena RF
6. Raspberry Pi 3B+ para centralizar la información de los tanques y para el envío de la información al servidor en la nube por HTTP.

- **Interfaces del comunicación** En la Tabla 1, se muestran las interfaces de comunicación entre los principales componentes del sistema de monitoreo.

Tabla 1. Interfaces de comunicación del sistema de monitoreo.

Sensores → Lora32 TTGO	I2C, OneWire
Lora32 TTGO → Nodo central	LoRa
Nodo central → Servidor	HTTP
Servidor → Nodo central	MQTT
Ciente → Servidor	HTPP

■ Interfaces de software

Los recursos de software seleccionados en cada componente para el desarrollo del sistema de monitoreo se muestran en la Tabla 2.

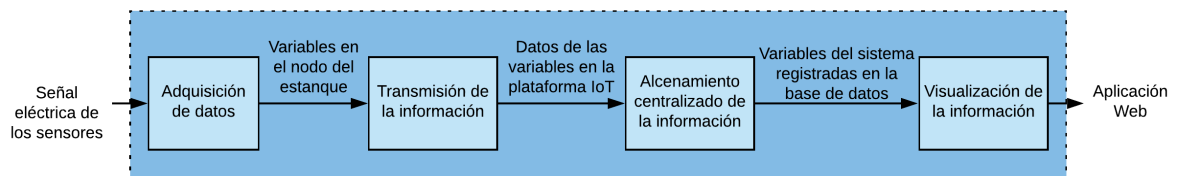
Tabla 2. Requisitos de software para el sistema de monitoreo.

Nodo en estanque	Librerías IDE Arduino	OneWire, Wire, SPI, DallasTemperature, LoRa
Nodo central	Librerías Python	paho.mqtt.client, json, serial, requests
Plataforma IoT Centro TIC	Sistema Operativo	Linux 4.15.0-32-generic, Ubuntu 16.04
	Base de datos	PostgreSQL 10
	Lenguaje servidor(backend)	Python3
	Lenguaje aplicación web (frontend)	HTML5, CSS, Javascript
	Librerías Python	scipy, argon2-cffi, bcrypt, dj-database-url dj-static, Django v1.11.5, django-bootstrap3, django-chartjs, Pillow, django-toolbelt, matplotlib, numpy, paho-mqtt, pandas, psycopg2, psycopg2-binary
	Servidor web	Apache2
	Broker MQTT	mosquito versión 1.4.8

3.1.5.2. Requisitos funcionales Los requisitos funcionales definen las acciones fundamentales que realizará el sistema, para el caso del sistema de monitoreo en estanques piscícolas basado en IoT, se muestra en la Fig. 4, la secuencia de estas acciones junto con sus entradas y salidas. Cada bloque representa una función del sistema y se describen a continuación:

- **Adquisición de datos:** En los estanques piscícolas se realizará un monitoreo en tiempo real de las variables con la frecuencia determinada.
- **Transmisión de la información:** Comprende el envío de la información desde el nodo en tanque, hasta la plataforma IoT del centro TIC.
- **Almacenamiento centralizado de la información:** La información de todos los estanques se almacenará en la base de datos del centro TIC.
- **Visualización de datos:** Los información se podrá visualizar en tiempo real desde cualquier dispositivo con acceso a internet.

Figura 4. Diagrama de bloques del Sistema de monitoreo en estanques piscícolas basado en IoT.



3.1.5.3. Requisitos no funcionales Los requisitos no funcionales describen características generales y restricciones del sistema, para este caso y teniendo en cuenta las necesidades de los actores, se presentan los siguientes requisitos no funcionales para el sistema de monitoreo:

- **Disponibilidad:** Se garantiza el acceso a la información de las variables del sistema de monitoreo debido al continuo funcionamiento del servidor web y de base d datos.
- **Escalabilidad:** El diseño general del sistema de monitoreo permite su crecimiento, debido a que cuenta con la capacidad de adaptar nuevos componentes para integrar nuevas funciones al sistema.
- **Seguridad:** La información de las variables transmitidas están protegidas contra posibles pérdidas en la transmisión ya que también se almacenaran en un micro SD en el terminal IoT.

3.1.5.4. Casos de uso Los casos de uso son eventos generales que describen todas las acciones posibles entre un actor y el sistema para un fragmento de funcionalidad dado. Una herramienta importante en la ingeniería de requerimientos es el diagrama de casos de uso el cual, es uno de los principales diagramas el Lenguaje Unificado de Modelado (UML); este tiene como objetivo describir las funciones proporcionadas por el sistema que produce un resultado visible para un actor ¹⁰.

Las Tablas 3, 4, 5, 6 presentan los principales casos de usos del sistema de monitoreo, posteriormente se presenta el diagrama de casos de uso en la Fig, 5, donde podemos observar la relación de los casos de uso del sistema de monitoreo con sus principales actores que son: los investigadores de los grupos de investigación Radiogis y Simon, el usuario registrado como administrador que pueden ser los profesores del Ipred quienes administran los estanques, y finalmente los usuarios sin registro.

¹⁰ B. BRUEGGE y A. DUTOIT. *Ingeniería de software orientada a objetos*. Pearson Education, 2002.

Tabla 3. Caso de uso: Almacenar la información.

Caso de uso	Almacenar la información
<i>Actores principales</i>	Radiogis, Simon
<i>Entrada</i>	Datos de cada uno de los estanques.
<i>Descripción</i>	Se realiza el almacenamiento centralizado de las variables medidas en todos los estanques.
<i>Salida</i>	Información registrada en la base de datos.

Tabla 4. Caso de uso: Visualizar variables.

Caso de uso	Visualizar variables
<i>Actores principales</i>	Radiogis, usuario administrador, usuario sin registrarse.
<i>Entrada</i>	Registro de la información en la base de datos.
<i>Descripción</i>	Los actores tienen acceso a visualizar la información del sistema de monitoreo por medio de una aplicación web. Dicha información también se puede descargar.
<i>Salida</i>	Interfaz gráfica de visualización de datos.

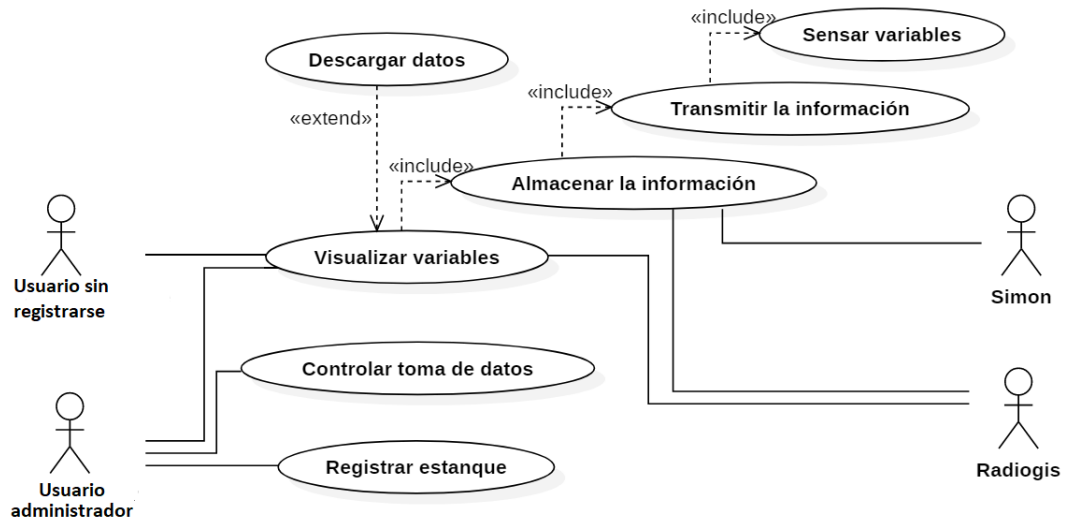
Tabla 5. Caso de uso: Controlar toma de datos.

Caso de uso	Controlar toma de datos
<i>Actores principales</i>	Usuario administrador
<i>Entrada</i>	Interacción con los botones ACTIVAR/DESACTIVAR de la pestaña <i>Monitoreo</i> de la aplicación web de Piscicultura.
<i>Descripción</i>	El usuario administrador es el único con permisos para activar y desactivar la toma de datos del sistema.
<i>Salida</i>	Activación/desactivación de la toma de datos del sistema de monitoreo.

Tabla 6. Caso de uso: Registrar estanque.

Caso de uso	Registrar estanque
<i>Actores principales</i>	Usuario administrador.
<i>Entrada</i>	Datos del estanque a registrar.
<i>Descripción</i>	El usuario administrador puede registrar nuevos estanques al sistema de monitoreo por medio de la pestaña <i>Registrar estanque</i> de la aplicación web.
<i>Salida</i>	Registro del estanque en la base de datos.

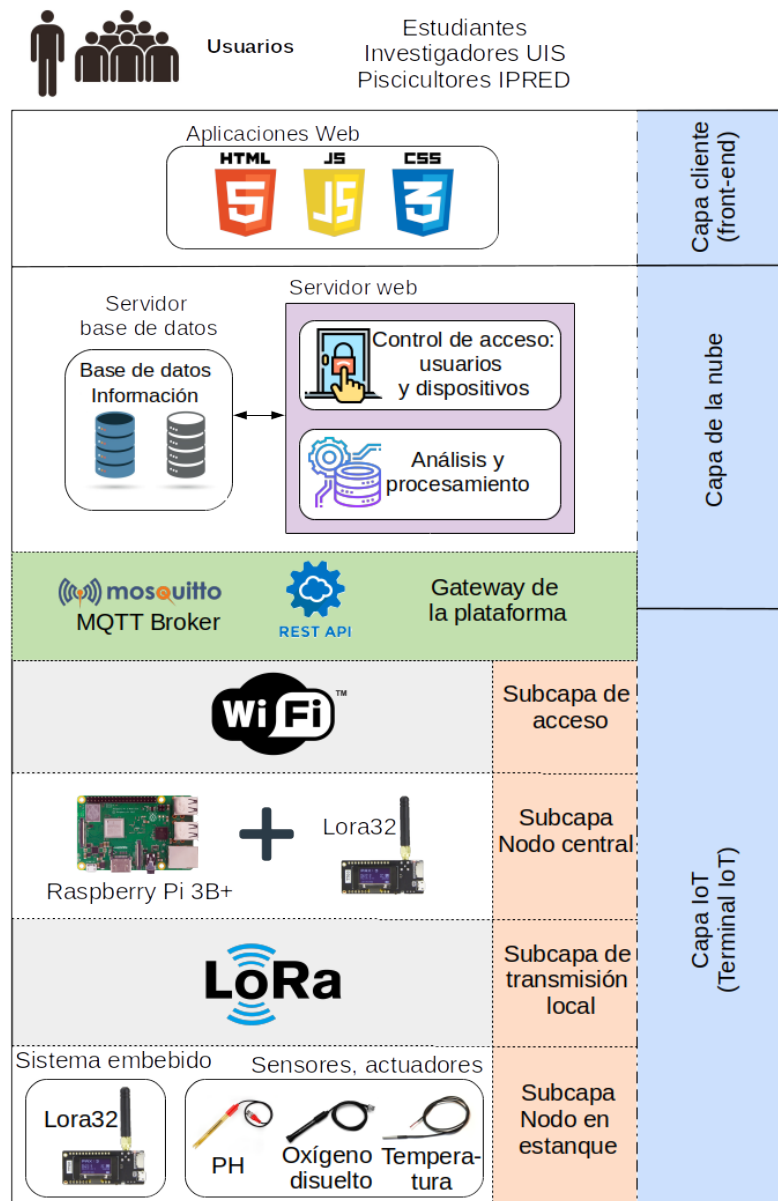
Figura 5. Diagrama de los Casos de uso para los actores del sistema.



3.2. PROCESO DE DISEÑO

3.2.1. Modelo general del sistema de monitoreo El modelo planteado para el sistema de monitoreo en estanques piscícolas basado en IoT, se muestra en la Fig. 6. Este modelo es una adaptación particular del modelo de capas para sistemas piscícolas inteligentes que se propuso en la sección 2.3. Este modelo presenta los componentes e interfaces que se usan en cada capa para la implementación de esta solución.

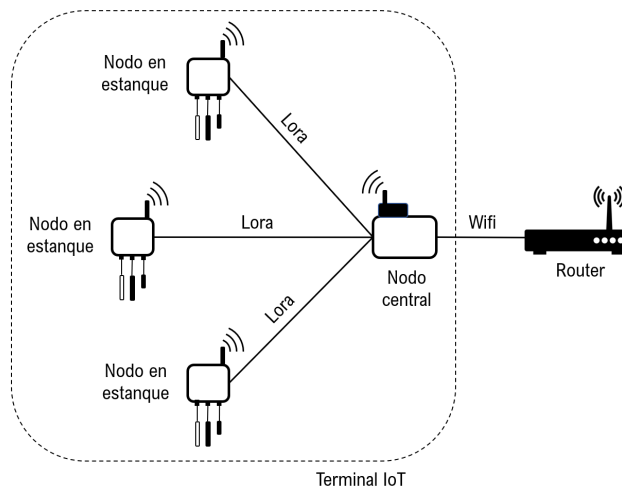
Figura 6. Modelo general del sistema de monitoreo en estanques piscícolas basado en IoT.



3.2.2. Red del Terminal IoT En la Fig. 7 se muestra el diagrama de red planteado para implementar en el terminal IoT, la red presentada se basa en la topología estrella que consiste en varios nodos conectados a un solo nodo central. Esta topología

fue escogida debido a que es más fácil proporcionar internet únicamente al nodo central y no a todos los nodos ubicados en los estanques, porque estos pueden estar en lugares de difícil acceso a internet y a su vez, es complicado garantizarles cobertura a todos al tiempo.

Figura 7. Red del terminal IoT.



Luego, para escoger la tecnología de transmisión a usar entre los nodos en estanque y el nodo central, se realizó la Tabla 7, que nos muestra una comparación general de las principales tecnologías usadas actualmente en aplicaciones de redes inalámbricas, con el fin de poder identificar la más adecuada de acuerdo a los requisitos del proyecto.

De acuerdo a la información mostrada en la tabla se opta por enviar la información hasta el nodo central por medio de LoRa ya que esta tecnología nos proporciona un rango del orden de los kilómetros, lo que puede ser muy útil en nuestra aplicación, para garantizar el internet al nodo central, ya que es muy posible que este tenga que estar ubicado lejos de los estanques, por lo tanto, es importante asegurar la transmisión en largas distancias. Además, esta tecnología soporta millones de dispositivos lo que le proporciona alta escalabilidad al sistema de monitoreo y su bajo consumo

Tabla 7. Comparación de tecnologías de comunicación.

Parámetro	Wifi	Bluetooth	LoRa	Xbee
Estándar IEEE	802.15.4	802.15.4	802.15.1	802.11
Rango	20-100m	10/100m	2-5km(urbano), 10-15km(rural)	20-100m
Frecuencia	5/2.4 GHz	2.4/6.9 GHz	915/868/430 MHz	915/868/2450 MHz
No. de dispositivos	256	8	Millones	2 a 65000
Consumo en transmisión	1	1	1	1
Consumo en reposo	1	1	1	1
Seguridad	WEP 64-128	Baja	AES 128 bits	AES 128 bits
Modulación	OFDM, QAM	GFSK	CSS	OQPSK

de energía lo hace altamente eficiente para aplicaciones IoT. A continuación se explica con más detalles, esta tecnología.

LoRa (Long Range) es una tecnología inalámbrica al igual que Zigbee, Wifi o Bluetooth, pero especialmente diseñada para IoT. Utiliza la modulación en radiofrecuencia conocida como Chirp Spread Spectrum (CSS) que fue patentada por Semtech una empresa que fabrica chips de radio. Algunas de las principales características de la tecnología LoRa son:

- **Largo alcance:** Hasta 5 km en zonas urbanas y 15 km en zonas rurales entre nodos sensores con bajas tasas de transferencia de datos que van de los 300 a los 50.000 bits por segundo.
- **Baja potencia:** Los nodos sensores pueden consumir muy poca energía y la batería durar años.
- **Bajo costo:** Los módulos de radio LoRa se pueden adquirir desde 3.5 dólares.

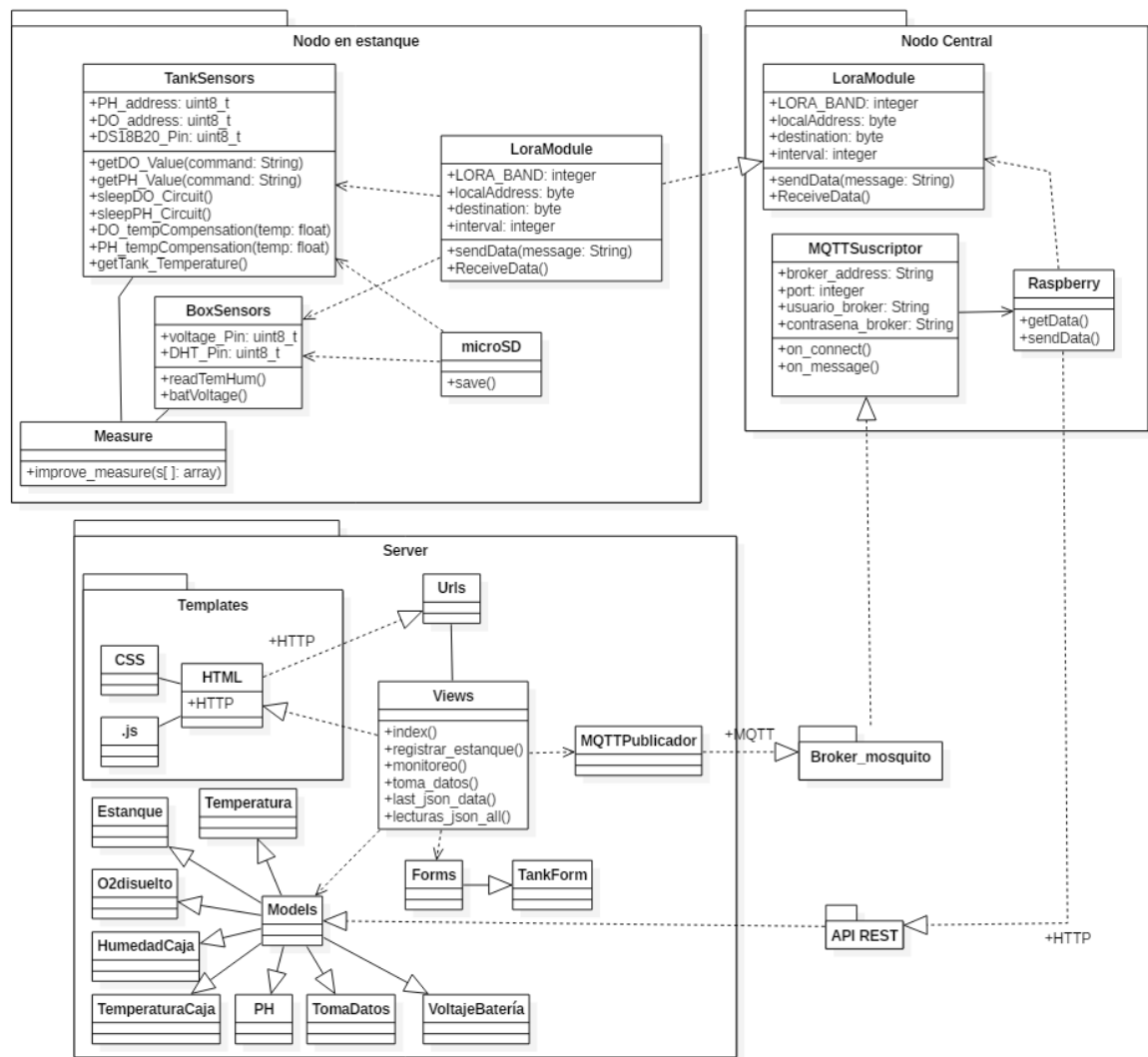
- **Seguridad:** Por su cifrado de datos AES-128 extremo a extremo integrado, es prácticamente imposible interceptar la información.
- **Altamente escalable:** Billones de sensores que pueden conectarse a millones de nodos.
- **Usa las bandas libres de frecuencia:** Lora usa el espectro sin licencia como parte de la banda de radio ISM (Industrial, Científica y Médica). 868 Mhz en Europa, 915 Mhz en América, y 433 Mhz en Asia.

3.2.3. Diagrama de clases Para el desarrollo del software de este proyecto se decide utilizar la programación orientada a objetos (POO). En los lenguajes orientados a objetos, los algoritmos definen *objetos* con atributos y métodos que pueden aplicarse a estos objetos, lo que permite modelar los objetos del mundo real. Por tanto, utilizar POO en este proyecto, permitirá que el software pueda ser fácilmente comprendido y modificado por un nuevo programador, para el posterior crecimiento del sistema.

Teniendo en cuenta lo mencionado, el diseño de la estructura software del sistema de monitoreo, se realiza utilizando el diagrama de clases que es una notación UML, el cual describe el sistema desde el punto de vista de objetos, clases, atributos, operaciones y sus asociaciones ¹⁰. En la Fig. 8 podemos ver el diagrama de clases correspondiente al software del sistema de monitoreo, las clases se representan como rectángulos que contienen sus atributos y operaciones, las líneas entre los rectángulos simbolizan las relaciones entre las clases que pueden ser de asociación, generalización, dependencia, entre otras. Este diagrama de clases del sistema, permite cumplir con los casos de uso planteados en la sección anterior y además, es una herramienta útil al momento de programar porque presenta toda la información importante de manera visual y concreta.

Este diagrama de clases esta conformado por tres paquetes, uno se ejecuta en el nodo en estanque, el segundo en el nodo centralizador y el tercero en el servidor de la plataforma del Centro TIC. La comunicación entre los nodos que hacen parte del terminal IoT se realiza por medio de la tecnología LoRa; la comunicación entre el nodo centralizador y el servidor se hace por medio de una petición POST con HTTP usando una interfaz *API_REST* y finalmente la comunicación servidor-nodo centralizador, se realiza con MQTT por medio de la interfaz *Broker_mosquito*.

Figura 8. Diagrama de los clases para el software del sistema de monitoreo en estanques piscícolas basado en IoT.

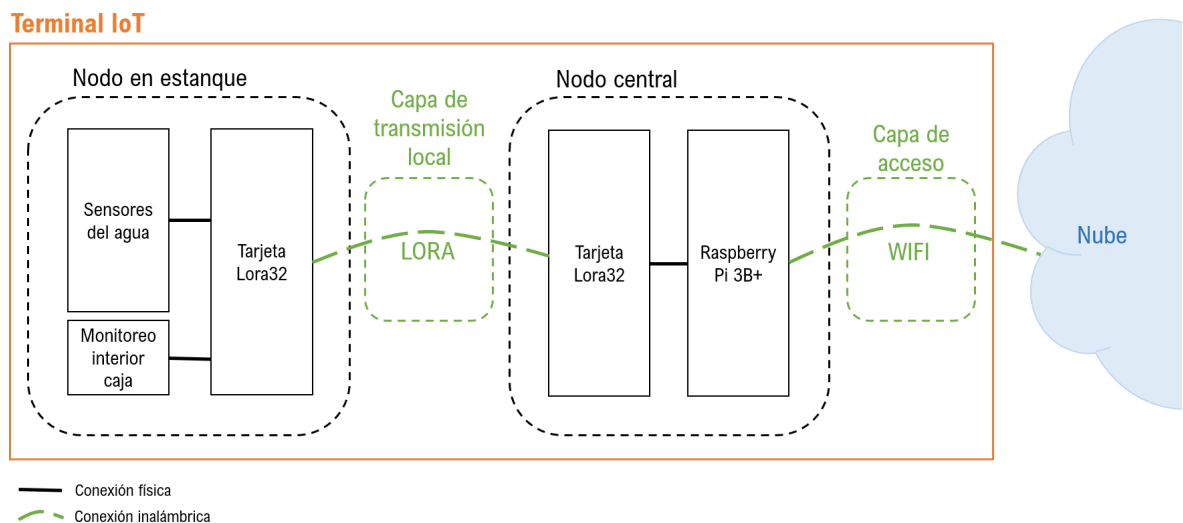


4. CAPA IOT

En este capítulo se presenta la siguiente etapa de la metodología para el desarrollo de soluciones inteligentes con IoT, la cual consiste en la implementación de la Capa IoT que se definió en el modelo del sistema de monitoreo en la Fig. 6. En esta capa se abarca todo lo relacionado con el terminal IoT y sus componentes. Se describe el desarrollo del nodo en estanque y el nodo central, la implementación de la subcapa de transmisión local y la subcapa de acceso.

El terminal IoT hace referencia a todos los elementos con capacidades de sensado y actuación, además, pueden conectarse a internet mediante la subcapa de acceso. Para este sistema de monitoreo el terminal IoT esta compuesto por varios elementos que se relacionan como se muestra en el diagrama de bloques de la Fig. 9.

Figura 9. Diagrama de bloques del terminal IoT.



4.1. NODO EN ESTANQUE

4.1.1. Sensores

Sensor de temperatura La temperatura del agua en piscicultura tiene gran influencia en el crecimiento y la actividad de los peces, ya que su temperatura corporal, es aproximadamente la misma del agua y varía con ella; por tal razón, es importante monitorear esta variable ¹¹.

Para la medición de la temperatura en el agua inicialmente se compararon las tres sondas mostradas en la Tabla 8. Luego, teniendo en cuenta que la temperatura en el agua del estanque piscícola no tomará valores negativos ni superaran los 40 °C, se decide escoger el sensor DS18B20 en presentación de sonda, el cual cuenta con el precio más económico y cumple con los requerimientos.

Tabla 8. Sensores de temperatura en líquidos.

Sensor	DS18B20	PT100	PT1000
<i>Rango [°C]</i>	-50 a 125	-20 a 450	-200 a 850
<i>Precisión [°C]</i>	+/- 0.5	+/- (0.15 + (0.002*t))	+/- (0.15 + (0.002*t))
<i>Salida</i>	Digital	Analógica	Analógica
<i>Sumergible 24/7</i>	Si	Si	Si
<i>Precio [COP]</i>	6.400	23.800	60.000

El sensor DS18B20 mostrado en la Fig. 10 es un dispositivo creado por la compañía *Maxim Integrated*, con la capacidad de comunicarse digitalmente con un microcontrolador por medio del protocolo de datos OneWire que utiliza un sólo hilo para la

¹¹ FAO. *Mejora de la calidad del agua en los estanques*.

transmisión de la información a diferencia de otros protocolos que usan más hilos

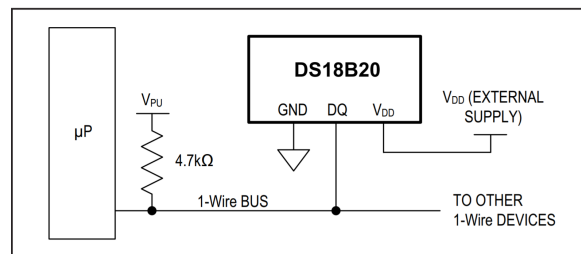
Figura 10. Sensor DS18B20.



Fuente: TIENDATEC. Disponible en: <https://www.tiendatec.es/148-sensores>

El DS18B20 puede ser usado en modo *Parásito de energía*, en el cual se utilizan sólo 2 cables: cable de tierra y el cable de datos del cual también toma la energía para funcionar. Sin embargo, para este caso se decidió conectar el DS18B20 utilizando los 3 terminales: tierra, datos y alimentación como se muestra en la Fig. 11.

Figura 11. Conexión sensor DS18B20.



Fuente: MAXIM INTEGRATED. DS18B20 - Programmable Resolution 1-Wire Digital Thermometer. Disponible en:

<https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>

Sensor de Ph El pH del agua de un estanque varía a lo largo del día principalmente a causa de la fotosíntesis de las algas en el estanque, y durante la noche a causa de la respiración. La producción de peces puede verse afectada por un pH demasiado bajo o alto. Por ejemplo, un pH inferior a 4.5 y superior a 11 causa la muerte de la mayor parte de los peces del cultivo. El pH en los rangos de 4.5-6.5 y

8.5-11 causa que el rendimiento sea muy pobre. Por otro lado, el agua cuyo pH varía entre 6.5 y 8.5 es la más adecuada para la producción de peces en estanques ¹¹.

Para medir el pH se compararon 3 opciones presentadas en la Tabla 9. La tercera opción a pesar de ser la opción más económica, la sonda no es apta para estar sumergida en el agua sensando continuamente, por tanto se descarta debido a que éste es uno de los requerimientos imprescindibles. Por otro lado, la primera y la segunda opción presentan características similares ya que son del mismo fabricante, pero la segunda a diferencia de la primera, incluye un sensor de temperatura para la corrección de la lectura de pH debido a que el pH varía en función de la temperatura. Sin embargo, ya se tiene un sensor de temperatura con un buen precio, por tanto, se elige la primera opción.

Tabla 9. opciones para medir pH en líquidos.

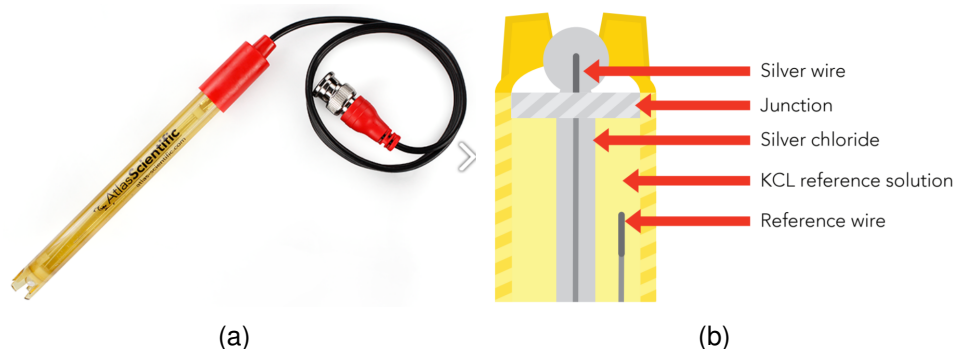
<i>Sensor</i>	Circuito sensor EZO y sonda pH Atlas Scientific	Circuito EZO pH y temperatura con sonda industrial pH Atlas Scientific	Circuito sensor y sonda pH DFRO-BOT
<i>Rango</i>	0 - 14	0 - 14	0 - 14
<i>Precisión</i>	+/- 0.002	+/- 0.002	+/- 0.1(25 °C)
<i>Tiempo de respuesta</i>	1 seg	1 seg	1 min
<i>Sensor temperatura</i>	No	Si (PT1000)	No
<i>Salida</i>	Digital	Digital	Analógica
<i>Sumergible 24/7</i>	Si	Si	No
<i>Precio [COP]</i>	345000	831000	151606

El sensor de pH de Atlas Scientific escogido, consta de una sonda pH y un circuito de pH EZO que se describirán a continuación.

■ ***Sonda de pH***

Esta sonda mide la actividad del ion hidrógeno en un líquido. Como podemos

Figura 12. Sonda de pH Atlas Scientific: (a) Electrodo para la medición de pH, (b) Partes del electrodo.



Fuente: 12.

ver en la Fig. 12(b), en la punta de la sonda hay una membrana de vidrio que permite que los iones de hidrógeno del líquido que se está midiendo se diluyan en la capa exterior del vidrio, mientras que los iones más grandes permanecen en la solución. La diferencia entre la concentración de iones de hidrógeno fuera de la sonda frente a la concentración medida por el electrodo de referencia en la solución de cloruro de potasio (KCL) dentro de la sonda, crea una corriente muy pequeña. Esta corriente es proporcional a la concentración de iones de hidrógeno en el líquido que se mide.¹²

El potencial que se desarrolla en el electrodo está determinado por la ecuación de Nernst¹³:

$$E = E_o + \frac{RT}{nF} \ln \alpha_H + = E_o - \frac{2.303RT}{F} pH \quad (1)$$

¹² ATLAS SCIENTIFIC. *pH Probe*. 2019.

¹³ La ecuación de Nernst se usa para calcular el potencial de un electrodo fuera de las condiciones estándares.

Donde:

E: Potencial eléctrico del electrodo de medida

E_0 : Potencial del electrodo de referencia, para obtener potencial 0 a pH=7.

R: Constante de los gases, $R= 31439.8 \text{ J/mol.K}$

T: Temperatura en Kelvin

n: Valencia iónica (valencia del hidrógeno $n=1$)

F: constante de Faraday, $F = 1.96493 \text{ C/mol}$

α_H : Cociente de reacción.

■ **Circuito pH EZO**

Este circuito es un sistema embebido que permite obtener fácilmente las medidas de la sonda de pH. Brinda gran precisión y resolución en las lecturas y efectúa la conversión de la señal analógica a digital. Además, soporta dos modos de comunicación: UART e I2C, por medio de los cuales entrega los datos al microcontrolador encargado de captar la información. Para este proyecto se decidió usar la comunicación por medio del protocolo I2C.

El circuito de pH EZO de Atlas Scientific mostrado en la Fig. 13 se calibra con 3 puntos (4.00, 7.00 y 10.00) los cuales le proporcionan una alta precisión en todo el rango de pH.

Figura 13. Circuito pH EZO.



Fuente: ATLAS SCIENTIFIC. Disponible en:
<https://www.atlas-scientific.com/ph.html>

Sensor de oxígeno disuelto Las necesidades de oxígeno de los peces están determinadas principalmente por la especie, el tamaño y la temperatura del agua. En algunas especies, los peces jóvenes necesitan niveles más altos de oxígeno que los adultos. En los períodos en que se alimentan activamente y luego durante la digestión, los peces necesitan más oxígeno que lo habitual. También consumen más oxígeno para respirar en aguas con temperaturas altas y cuando la temperatura aumenta el agua contiene menos oxígeno por tanto es importante conocer los niveles oxígeno en el agua del estanque para tomar medidas como encender la re-circulación solamente cuando sea necesario ¹¹.

Para la medición del oxígeno disuelto en el agua del estanque piscícola se compararon las dos opciones que se muestra en la Tabla 10. La opción presentada por el fabricante Atlas Scientific tiene mejores características de interés, como el tiempo de respuesta, vida útil en comparación que la opción de DFRobot. Por tanto, se decide usar la primera opción presentada en la tabla.

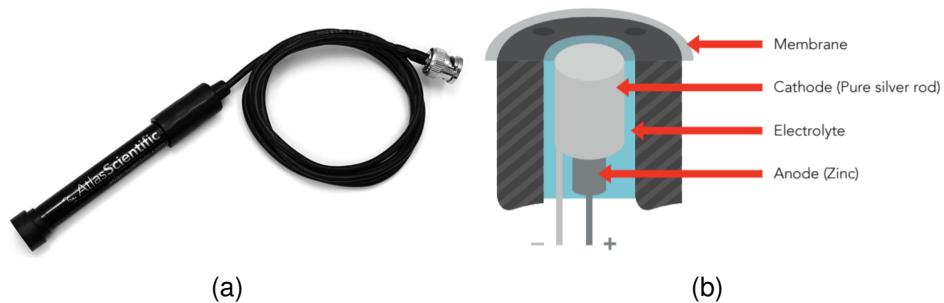
Tabla 10. Opciones para medir oxígeno disuelto en líquidos.

<i>Sensor</i>	Circuito sensor EZO y sonda Oxígeno Disuelto Atlas Scientific	Circuito y sonda Oxígeno Disuelto DFRobot
<i>Rango</i>	0 - 100 mg/L	0 - 20 mg/L
<i>Precisión</i>	+/- 0.05 mg/L	-
<i>Tiempo de respuesta</i>	1 seg	90 seg
<i>Salida</i>	Digital	Analógica
<i>Vida útil</i>	5 años	1 año
<i>Precio [COP]</i>	849552	833000

El sensor de oxígeno disuelto de Atlas Scientific escogido, consta de una sonda y un circuito EZO que se describirán a continuación.

■ ***Sonda de oxígeno disuelto***

Figura 14. Sonda de pH Atlas Scientific: (a) Sonda galvánica para la medición de oxígeno disuelto, (b) Partes de la sonda.



Fuente: 14.

Esta sonda de oxígeno disuelto se puede sumergir completamente en agua dulce o salada, hasta el conector BNC de forma indefinida.

Una sonda galvánica de oxígeno disuelto consiste en una membrana de polietileno, un ánodo bañado en un electrolito y un cátodo como se muestra en la Fig. 14(b). Las moléculas de oxígeno se desactivan a través de la membrana de la sonda a una velocidad constante (sin la membrana la reacción ocurre rápidamente). Una vez que las moléculas de oxígeno han cruzado la membrana, estas se reducen en el cátodo y se produce un pequeño voltaje. Si no hay moléculas de oxígeno presentes, la sonda emitirá 0 mV. A medida que aumenta el oxígeno, también lo hace la salida de mV de la sonda ¹⁴.

■ **Circuito oxígeno disuelto EZO**

Este circuito es un sistema embebido que entrega las lecturas de la sonda de oxígeno disuelto en digital. Este se controla por medio de comandos enviados desde el microcontrolador a través del protocolo UART o I2C. En este proyecto,

¹⁴ ATLAS SCIENTIFIC. *Dissolved Oxygen Probe*. 2019.

para la comunicación del circuito EZO de oxígeno disuelto se decidió usar el protocolo I2C al igual que con el circuito EZO de pH, lo cual permite conectar ambos circuitos a las mismas líneas SDA, SCL y por medio de las direcciones I2C de cada circuito EZO se diferencian ambos sensores.

El circuito de oxígeno disuelto EZO de Atlas Scientific mostrado en la Fig. 15, tiene un protocolo de calibración flexible, que permite un solo punto o punto dual (opcional) de calibración. Cuando se realiza con un solo punto se usan los niveles de oxígeno de la atmósfera, cuando se agrega un segundo punto de calibración se usa la solución de calibración de cero oxígeno disuelto.

Figura 15. Circuito OD EZO.



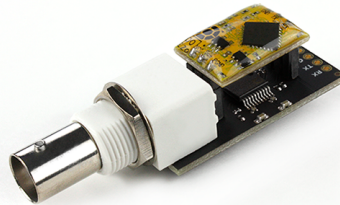
Fuente: ATLAS SCIENTIFIC. Disponible en:
<https://www.atlas-scientific.com/dissolved-oxygen.html>

Placa portadora EZO aislada eléctricamente

El circuito de oxígeno disuelto EZO y de pH de Atlas Scientific es un dispositivo muy sensible. Esta sensibilidad es lo que le da a los circuitos su precisión lo que significa que también es capaz de leer micro voltajes que se descargan en el agua de fuentes no naturales, como bombas, válvulas solenoides u otras sondas/sensores. Como en este caso vamos a leer pH y oxígeno disuelto al tiempo, es necesario aislar los circuitos EZO de pH y de oxígeno disuelto, por medio de la placa portadora EZO aislada eléctricamente que ese muestra en la Fig. 16. Esta ofrece una forma eficiente y fácil de conectar los circuitos EZO de Atlas Scientific con la sondas ya

que incluye un conector BNC macho, y en la misma placa aísla eléctricamente las lecturas.

Figura 16. Placa portadora EZO aislada eléctricamente.



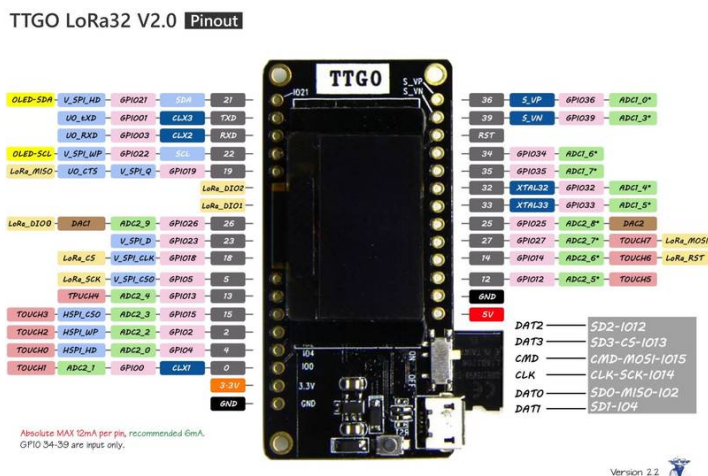
Fuente: ATLAS SCIENTIFIC. Disponible en: https://www.atlas-scientific.com/product_pages/components/single_carrier_iso.html

4.1.2. Tarjeta de desarrollo Para adquirir las lecturas de los sensores y luego transmitir inalámbricamente la información es necesario un microcontrolador. Para este proyecto se consideraron las tarjetas de desarrollo mostradas en la Tabla 11, las cuales traen un microcontrolador y otros periféricos para facilidad en la programación y en el uso. Las tarjeta escogida debe contar con entradas y salidas digitales, comunicación I2C, y tecnologías de transmisión de la información. Teniendo en cuenta que en la sección 3.2.2 se diseñó la red del terminal IoT y se definió usar Lora para la transmisión de datos desde el nodo en estanque, se decide trabajar con la tarjeta *LoRa32* del fabricante *TTGO* mostrada en la Fig. 17. Esta tarjeta de desarrollo es la única opción de la tabla que cuenta con comunicación Lora debido a que incluye un módulo de transmisión RF para este uso, además, cumple con las otras características requisito y adicionalmente cuenta con un lector de tarjetas microSD que puede ser útil para almacenar un respaldo de la información.

Tabla 11. Tarjetas de desarrollo

<i>Tarjeta</i>	Lora32 V2.0	TTGO	Esp32 Dev Kit	Arduino nano
<i>Microcontrolador</i>	Esp32		Esp32	Atmega328
<i>Voltaje de operación [V]</i>	3.3 - 5		3.3 - 5	5
<i>Comunicación</i>	UART, SPI, I2C		UART, SPI, I2C	UART, SPI, I2C
<i>I/O Digitales</i>	24		25	14 (6 son PWM)
<i>Entradas Analógicas</i>	16		15	8
<i>Conectividad</i>	Wifi, Bluetooth, Lora		Wifi, Bluetooth	-
<i>Precio [COP]</i>	73000		64000	23000

Figura 17. Pines tarjeta de desarrollo Lora32 de TTGO.



Fuente: BRICOLABS. LoRa + TTN: comunicando cosas con Internet. Disponible en: https://bricolabs.cc/wiki/guias/lora_ttn

Módulo Lora La tarjeta de desarrollo Lora32 de TTGO incluye el módulo de transmisión Lora SX1276, este se comunica con el Esp32 por medio del protocolo SPI y permite la comunicación en la frecuencia de 915 MHz.

Los transceptores SX1276/77/78/79 cuentan con el módem de rango largo LoRaTM, que proporciona comunicación de espectro ensanchado de alcance ultra-largo que supera el rango de los sistemas existentes basados en la modulación FSK u OOK. Además, proporciona alta inmunidad a interferencias mientras minimiza el consumo de corriente. A continuación se muestran las principales características:

- Alta sensibilidad: hasta -148 dBm.
- Velocidad de bits programable hasta 300 kbps.
- Modulaciones: FSK, GFSK, MSK, GMSK, LoRaTM, OOK.
- Voltaje de operación: 1.8-3.7 V
- Corriente de alimentación en modo Sleep: 0.2 uA.
- Ancho de banda con modulación LoraTM: 125 kHz.
- Potencia de salida modulo LoraTM: 13 dB, en transmisión.
- Corriente en transmisión: 28 mA.
- Corriente en recepción: 11.5 mA.

4.1.3. Fuente de alimentación Para este proyecto se decidió utilizar una batería recargable de litio 18650 como la que se muestra en la Fig. 18 . Esta batería tiene un voltaje de 3.7 [V] y una capacidad nominal de 4000 [mA].

Figura 18. Batería de litio 18650.

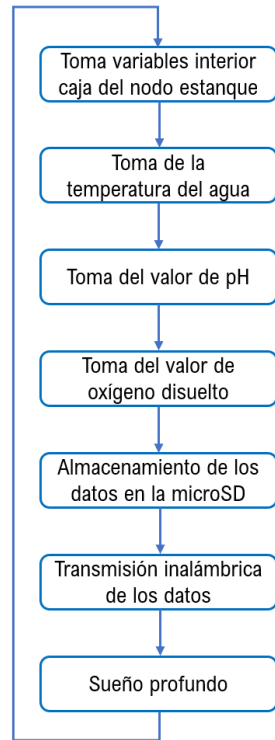


4.1.4. Descripción del funcionamiento El código cargado a la tarjeta de desarrollo Lora32 del nodo en estanque realiza periódicamente ciertas acciones que se muestran en el diagrama de la Fig. 19. El ciclo inicia con la toma de variables dentro de la caja, posteriormente, se capturan los valores de las variables del agua como la temperatura, el pH y el oxígeno disuelto. Es importante mencionar que para obtener un valor preciso de las variables, es necesario aplicar un pre-procesamiento que consiste en tomar 5 medidas de las variables, ordenarlas de menor a mayor, descartar el menor y el mayor valor. Con esto, se garantiza que no se afecte el valor final a causa de errores en alguna medida. Luego, con las medidas restantes se hace un promedio cuyo resultado es el valor final de la variable ¹⁵.

Luego de obtener el resultado final de cada variable, continuando con el ciclo, se procede a almacenar esta información en la microSD que será un respaldo de los datos, útil caso de alguna falla en la transmisión inalámbrica. Finalmente el nodo en estanque entra en modo sueño profundo por un tiempo determinado configurado previamente. Este modo es útil porque permite ahorrar batería mientras no se toman datos de las variables. Luego, inicia el ciclo nuevamente.

¹⁵ Namir EL-KHOURI VIDARTE. *Adaptación e implementación de un sistema autónomo de bajo coste de monitorización de calidad del agua en tiempo real*. Trabajo de Grado de ingeniería de tecnologías y sistemas de telecomunicación. Madrid- España: Universidad Politécnica de Madrid, 2016.

Figura 19. Diagrama de funcionamiento del nodo en estanque.



4.2. SUBCAPA DE TRANSMISIÓN LOCAL

La implementación del enlace entre el nodo en estanque y el nodo central se hace por medio de la tecnología de transmisión Lora como se decidió en la sección 2.2.2 donde se realizó en diseño de la red del Terminal IoT.

4.2.1. Datos enviados Desde el nodo en estanque se envía la siguiente trama de datos con las variables separadas por comas en el siguiente orden:

id_,temp_measure,ph_measure,do_measure,humBox_measure,tempBox_measure,volt_bat

Donde:

- *id_*: Es el identificador del estanque.
- *temp_measure*: Es la temperatura del agua.
- *ph_measure*: Es el pH del agua.
- *do_measure*: Es el oxígeno disuelto en el agua.
- *humBox_measure*: Es la humedad dentro de la caja del nodo en estanque.
- *tempBox_measure*: Es la temperatura dentro de la caja del nodo en estanque.
- *volt_bat*: Es el voltaje en la batería.

El nodo central recibe la trama y la Raspberry se encarga de identificar las comas y separar los datos de las variables para su posterior envío a internet.

Para mejorar aun mas la robustez del enlace LoRa, el módulo LoRaTM emplea chequeo de redundancia cíclica (CRC - Cyclic Redundancy Check) y corrección de errores hacia adelante (FEC-Forward Error Correction). Esta codificación del error implica un incremento en pequeño incremento en los datos transmitidos.

4.3. NODO CENTRAL

Este componente del terminal IoT se encarga de centralizar la información de todos los estanques con monitoreo en la misma zona, para luego enviarla a la plataforma IoT en la nube y poder visualizar la información de los estanques junta. A continuación se describen los elementos que hacen parte del nodo central.

4.3.1. Módulo receptor Para recibir la información de los estanques con monitoreo, se decidió usar como módulo receptor la tarjeta de desarrollo Lora32 mencionada en la sección 3.1.1.2, debido a su capacidad de comunicación por la tecnología inalámbrica Lora.

4.3.2. Raspberry Pi 3B+ La Raspberry Pi 3B+ que se muestra en la Fig. 20, es un computador de bajo costo, que se adapta múltiples aplicaciones según los requerimientos del usuario. En la Tabla 12 se muestran las principales características de la Raspberry Pi 3B+. En este caso, la Raspberry tiene la función de adquirir los datos recibidos por el módulo receptor (Lora32), para luego enviarlos a la plataforma en la nube. Además, debe recibir ciertas acciones que el cliente activa desde la aplicación web.

Figura 20. Raspberry Pi 3B+



Fuente: RASPBERRY PI. Disponible en:

<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>

Tabla 12. Características Raspberry Pi 3B+.

<i>CPU+GPU</i>	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
<i>RAM</i>	1GB LPDDR2 SDRAM
<i>Wifi</i>	2.4GHz y 5GHz IEEE 802.11.b/g/n/ac
<i>Bluetooth</i>	Bluetooth 4.2, BLE
<i>Ethernet</i>	Gigabit Ethernet sobre USB 2.0 (300 Mbps)
<i>GPIO</i>	40
<i>Puertos USB 2.0</i>	4

4.3.3. Descripción del funcionamiento La tarjeta de desarrollo *Lora32* que funciona como módulo receptor en el nodo central, realiza las siguientes acciones continuamente:

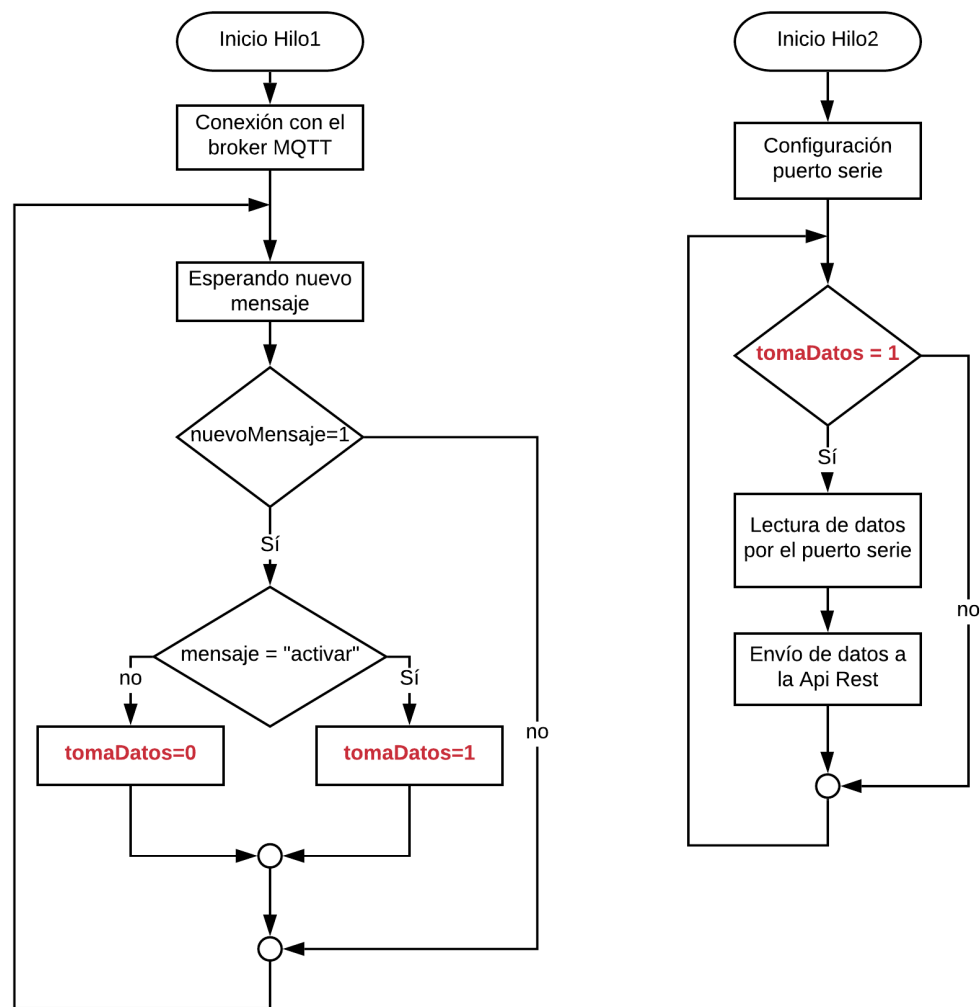
- Recepción de datos inalámbricamente por Lora.
- Envío de datos por medio del puerto serie.

Por otro lado, en la Raspberry se ejecuta un código en Python para el cual fue necesario implementar programación en dos hilos, debido a que no fue posible mantener continuamente el estado en espera de mensajes provenientes de la aplicación web, mientras al mismo tiempo se realizaban lecturas del puerto serie. Esta característica es importante si queremos pausar la toma de datos en la Raspberry desde la aplicación web, por tanto se implementó la solución de programar un hilo para cada proceso de una manera fácil, con la ayuda de la librería *threading* que se importa en Python para este uso.

En la Fig. 21 podemos ver los diagramas de flujo de los dos procesos implementados. El hilo 1 corresponde al estado continuo en espera de mensajes y el hilo 2 hace referencia a la lectura del puerto serie y el envío de datos a internet. Lo más importante a resaltar en el hilo 1 es que cuando llega un mensaje, dependiendo de

su contenido, la variable global *tomaDatos* modifica su valor para indicarle al hilo 2 cuando debe activar la toma de datos en la Raspberry.

Figura 21. Diagrama de flujo de los dos hilos ejecutados en la Raspberry.



4.4. SUBCAPA DE ACCESO

La incorporación de esta subcapa en los componentes del sistema, es la característica más importante para poder llamarlos terminales IoT. Entre las tecnologías

usadas en la subcapa de acceso se encuentran: WiFi/IEEE802.11, redes móviles 3G/4G/5G, LTE/LTE-A, ZigBee, entre otros estándares 802.15.4. Las conexiones cableadas también hacen parte de esta subcapa, como lo es el Ethernet. Sin embargo, para este proyecto los terminales IoT acceden a internet por medio de la interfaz de comunicación **Wifi IEEE802.11**.

5. GATEWAY DE LA PLATAFORMA

En este capítulo se presentan los recursos usados en la implementación de este proyecto para la gestión de dispositivos como el Broker MQTT y la Api Rest, los cuales son esenciales en la comunicación entre el terminal IoT y la plataforma IoT en la nube.

5.1. API REST

Una API (*Application Programming Interfaces*) es un conjunto de reglas y especificaciones que facilita las interacciones entre diferentes softwares para permitirles intercambiar información. Una API web es la cara de un servicio web, que escucha y responde directamente a las solicitudes. El estilo arquitectónico REST (*Representational State Transfer*) se aplica comúnmente al diseño de APIs para servicios web modernos. Una API web que se ajuste al estilo arquitectónico REST es una API REST ¹⁶.

Características de una API REST

- Los objetos en REST son manipulados con la URL.
- Trabaja con operaciones importantes para realizar un CRUD (*Create Read Update Delete*) como: POST, GET, DELETE, PUT.
- Separación entre el cliente y el servidor, mejora la portabilidad de la interfaz a otro tipo de plataformas.

¹⁶ M. MASSÉ. *REST API Design Rulebook*. O' Reilly Media, 2012.

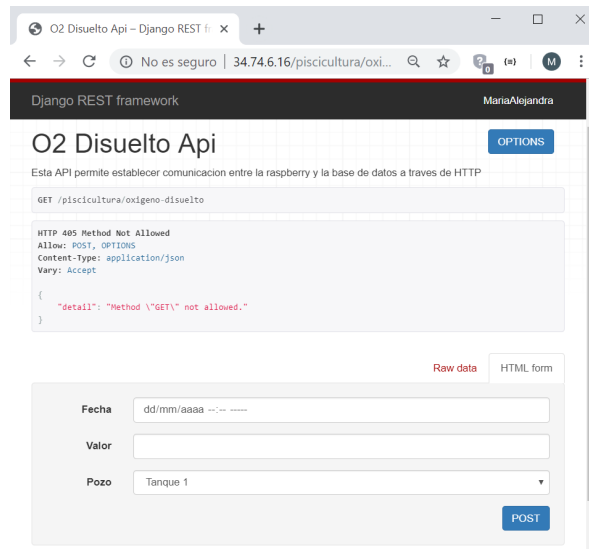
- Visibilidad, fiabilidad y escalabilidad. Permite migrar a otros servidores y realizar todo tipo de cambios en la base de datos con el envío correcto de las peticiones.
- La API REST es independiente del tipo de plataformas se pueden tener servidores en PHP, Python, Java o Node.js, las peticiones siempre se deben realizar utilizando el lenguaje de intercambio XML o JSON.

Envío de datos desde el Terminal IoT a la Api Rest

La plataforma del Centro TIC tiene preestablecido el uso de una Api Rest para la comunicación desde el Terminal IoT hacia la plataforma, debido a que es el método más práctico y utilizado para enviar datos a Internet por medio del protocolo HTTP(*HyperText Transfer Protocol*). En este proyecto, utilizamos API REST por medio de *django REST framework*, el cual es un kit de herramientas potentes y flexibles para crear APIs web y que se instala en el servidor. Esta herramienta, cuenta con amplia documentación y gran apoyo de la comunidad ¹⁷. Una ventaja grande del uso de API REST junto a *django* es que nos permite por medio de la autenticación con Tokens controlar que dispositivos están autorizados para enviar datos a la API. Fue necesario crear una API para cada variable monitoreada para realizar el envío de datos. En la Fig. 22 podemos ver por ejemplo, la APIView de la API creada para en envío de datos de la variable oxígeno disuelto.

¹⁷ MKDOCS. *Django REST framework*. 2011.

Figura 22. Vista de la API creada para la variable oxígeno disuelto



El envío de las datos desde el Terminal IoT a la API instalada en el servidor, se realizó por medio del protocolo HTTP, utilizando el **método POST** desde el terminal IoT. Este método es utilizado para enviar una entidad a un recurso en específico, causando un cambio en el estado o efectos secundarios en el servidor. En este caso, enviamos la información de las variables de la plataforma del Centro Tic.

En la Raspberry del Terminal IoT, realizamos la petición POST con la función *requests.post()* como se muestra a continuación:

```
r = requests.post(url, data=payload, headers={"Authorization":  
"Token c9f16d59ffc9c48d24c0c694b10970110d47ccf4"})
```

Donde:

- El primer parámetro es la URL de la solicitud, que equivale a la URL asignada a la *APIView* de cada variable.
- El parámetro *data* hace referencia a la información a enviar, en este caso se

envía un diccionario con tres pares {clave: valor} como se muestra a continuación:

```
payload = {  
    "fecha": date ,  
    "valor": variable ,  
    "pozo": id_  
}
```

- El parámetro *headers* se refiere a un diccionario de encabezados HTTP para enviar con solicitud. En este caso se envía un diccionario con un solo par {clave: valor}, el cual es importante para la identificación del usuario que esta enviando datos, para el caso de APIs con permisos de uso para ciertos usuarios.

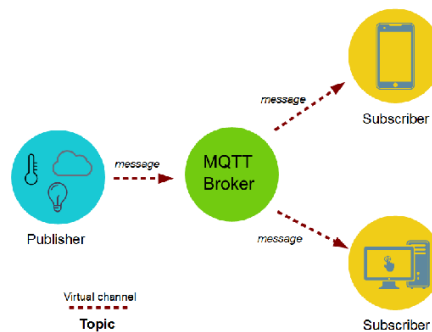
5.2. BROKER MQTT MOSQUITO

Por otro lado, el Broker MQTT *Mosquito* es el encargado de gestionar la comunicación desde la plataforma IoT del Centro TIC hacia el Terminal IoT. Este recurso nos permite enviar de un manera fácil mensajes a los Terminales IoT incluso a los más sencillos, ya que no necesita la ejecución de un servidor para recibir los datos. Además, esta herramienta se viene utilizando en Radiogis hace un largo tiempo, por lo cual se ha adquirido experiencia en su uso. Para este proyecto, este recurso permite enviar al nodo central la acción de ACTIVAR/DESACTIVAR la toma de datos que el usuario controla por medio de dos botones en la aplicación web. El Broker Mosquito utiliza MQTT (*Message Queue Telemetry Transport*), el cual es un protocolo ligero de conectividad IoT/M2M (*Machine to machine*) y fue diseñado como un transporte

de mensajería basado en publicador/suscriptor ¹⁸.

El publicador envía el mensaje al *Broker* a través de un tópico(identificador de la comunicación con estructura jerárquica, que se define mediante un *string*), luego, los dispositivos que estén suscritos al tópico son lo que recibirán el mensaje del publicador como se muestra en la Fig. 23.

Figura 23. Diagrama de funcionamiento MQTT



Fuente: SKYFILABS. IoT based Real time data acquisition using MQTT Protocol.
Disponible en: <https://www.skyfilabs.com/project-ideas/iot-based-real-time-data-acquisition-using-mqtt-protocol>

Para utilizar el protocolo MQTT en esta solución, fue necesario lo siguiente:

1. Se definió el tópico “*UIS/PISCICULTURA*” para la comunicación.
2. En el terminal IoT, se instalaron las librerías del suscriptor MQTT, el cual estaba suscrito al tópico que se mencionó en el ítem anterior.
3. En el servidor web en la nube, está el broker MQTT el cual es el elemento principal en la gestión de los dispositivos en MQTT, se encarga de mantener la comunicación y redireccionar los mensajes del servidor web con el terminal

¹⁸ OASIS MQTT. *MQTT Version 3.1.1*. 2014.

IoT. Además, se instalaron las librerías del publicador MQTT para enviar los mensajes desde la plataforma del Centro TIC.

6. CAPA DE LA NUBE

En este capítulo se desarrolla la siguiente etapa de la metodología para soluciones IoT, que hace referencia a los componentes de la solución implementados en la nube que permiten el almacenamiento de la información, el procesamiento y el control de acceso a usuarios y dispositivos. Por lo tanto, aquí se explica de manera general el modelo de datos usado para almacenar la información en la base de datos, un sencillo ejemplo de procesamiento y los métodos usados para identificarse en el sistema.

6.1. ALMACENAMIENTO EN LA BASE DE DATOS

La información de las variables monitoreadas, luego de ser enviadas a la API Rest, se guardan en una base de datos implementada en postgresSQL con ayuda del *framework* web Django utilizado en la plataforma del Centro TIC. Esta base de datos sigue un modelo relacional el cual proporciona una estructura de los datos que consiste en un conjunto de relaciones con objeto de representar la información que nos interesa del mundo real ¹⁹. En la Fig. 24 se observa el modelo de datos implementado donde se observan principalmente tres tipos de tablas:

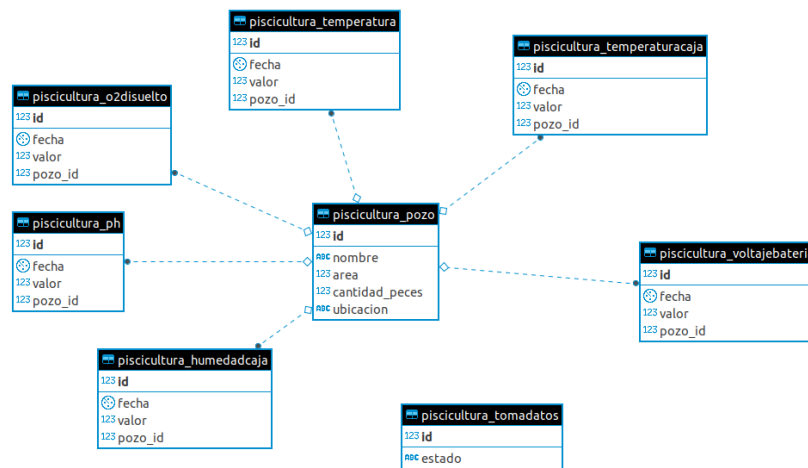
- Una tabla de información de los estanques llamada *piscicultura_pozo* con los atributos *nombre*, *área* y *cantidad de peces*.
- Una tabla para cada variable con los atributos: *fecha*, *valor* y *pozo_id*. Estos campos corresponden a las claves del diccionario enviado por el método POST

¹⁹ Luis Alberto CAMPS PARÉ Rafael y CASTILLAS SANTILLÁN. *Bases de datos*. Universitat Obrera de Catalunya, 2005.

a la API Rest. El atributo *pozo_id* es una clave foránea²⁰ que permite que esta tabla se relacione con la tabla *piscicultura_pozo*.

- Una tabla *piscicultura_tomados* con un atributo llamado *estado*, donde se guarda el estado de la toma de datos accionado desde la aplicación web.

Figura 24. Modelo de la base de datos implementada en postgresSQL



6.2. EJEMPLO DE PROCESAMIENTO EN LA NUBE

Para este sistema de monitoreo, se implemente un procesamiento sencillo en la nube, que corresponde a encontrar el valor promedio de las últimas 18 lecturas de las variables del agua. Por ejemplo, si se configura el terminal IoT para que envíe datos cada 10 minutos, entonces se promediaran las lecturas de las últimas 3 horas.

²⁰ Es una columna de una tabla que contiene valores que coinciden con la clave primaria de otra tabla

6.3. CONTROL DE ACCESO A USUARIOS Y DISPOSITIVOS

El *framework* Django usado en la plataforma del centro TIC, ofrece un sistema de autenticación de usuarios, que maneja la autenticación y la autorización juntas. La autenticación verifica que un usuario es quien dice ser, y la autorización determina lo que un usuario autenticado puede hacer ²¹.

Para este proyecto se creó el usuario *AdminEstanques*, el cual es el único con permisos para registrar nuevos estanques y enviar datos desde el terminal IoT. Para el acceso al formulario de registro de nuevos estanques la autenticación se realiza ingresando el usuario y la contraseña. En el terminal IoT la autenticación en la API Rest se hace por medio de un *token*²² asignado al usuario *AdminEstanques*.

²¹ DJANGO SOFTWARE FOUNDATION. *Django Documentation*. 2019.

²² Cadena de caracteres que identifican al usuario en el lado del servidor

7. CAPA CLIENTE

En este capítulo se presenta la penúltima etapa de la metodología para el desarrollo de soluciones IoT, que consiste en el desarrollo de la capa cliente. De acuerdo a los requisitos de las interfaces del usuario definidos en la sección 2.1.5.1. en este capítulo se explicará brevemente cómo se llevó a cabo cada una de ellas, las cuales hacen parte de la aplicación web con la que interactúa el usuario.

Antes de entrar en detalle de cada una de las vistas implementadas para la aplicación web del sistema de monitoreo en estanques piscícolas, es importante tener una noción básica de las herramientas utilizadas que fueron las siguientes:

- **HTML (*Hypertext Markup Language*):** Es el lenguaje de marcado de texto utilizado para la creación de páginas web. Los comando de HTML le indican al software del navegador la estructura del documento y, cuando corresponda, cómo desea que se muestre el contenido ²³.
- **CSS (*Cascading Style Sheets*):** Es una tecnología que permite de manera sencilla personalizar la presentación visual de los elementos escritos en el lenguaje HTML. CSS abarca desde esquinas redondeadas y sombras hasta transformaciones y reposicionamiento de los elementos presentados en pantalla ²⁴.
- **JavaScript:** Es un lenguaje complementario, en la mayoría de casos con HTML y CSS para mejorar la web. Esta tecnología se ejecuta en el lado del cliente y

²³ Thomas A. POWELL. *HTML: The Complete Reference*. McGraw-Hill, 1999.

²⁴ Juan D. GAUCHAT. *El gran libro de HTML5, CSS3 y Javascript*. Marcombo, 2012.

puede ayudar a convertir una página estática de contenido en una experiencia atractiva, interactiva e inteligente ²⁵.

La aplicación de piscicultura en la plataforma IoT del Centro TIC se encuentra en el siguiente link: <http://35.243.199.245/piscicultura/index>, allí en la parte superior se encuentran tres pestañas que permiten acceder a las vistas descritas a continuación:

7.1. REGISTRO DE ESTANQUES

Para la elaboración de la vista mostrada en vista de la Fig. 25(a) se usó HTML5 y CSS3. Además, se implementó un formulario para el registro de nuevos estanques junto a toda su información. Esto se realizó con ayuda de Django, que nos proporciona una variedad de herramientas y bibliotecas para la creación de formularios de una manera sencilla y rápida ²⁶.

Con el método POST, el navegador agrupa los datos del formulario, los codifica para su transmisión y los envía a la base de datos para el registro en la tabla *piscicultura_pozo*. Para poder acceder a esta vista, es necesario que el usuario *AdminEstanques* inicie sesión debido a que es el único con permiso para registrar nuevos estanques. En la Fig. ?? podemos ver la vista para el registro de estanques.

7.2. TOMA DE DATOS

Es esta vista, solo el usuario *AdminEstanques* puede activar o desactivar la toma de datos mediante los botones de ACTIVAR o DESACTIVAR como se puede ver en la

²⁵ Danny GOODMAN. *JavaScript Bible*. Hungry Minds, Inc., 2001.

²⁶ DJANGO. *Working with forms*. 2019.

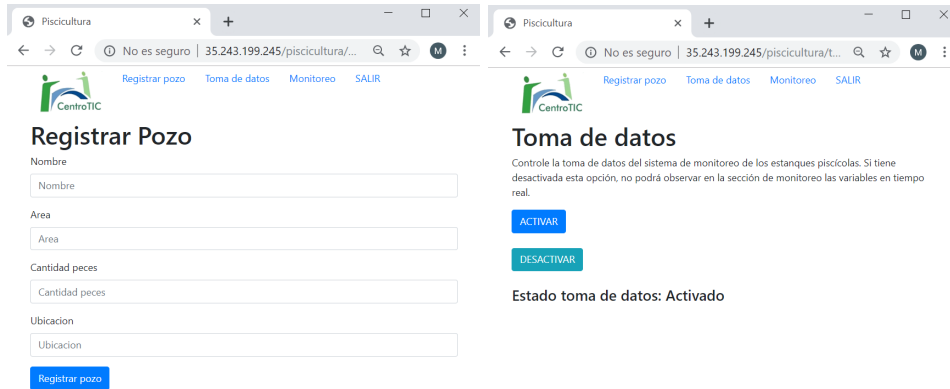
Fig. 25(b). Para la lectura del accionamiento de estos botones, es necesario utilizar Javascript, una vez ocurre el evento de *click* en alguno de estos, se envía un POST al servidor en formato JSON (*JavaScript Object Notation*) ²⁷ con la información de activar o desactivar según corresponda. Luego, la información de la orden es enviada por medio de MQTT al terminal IoT, el cual siempre está a la espera de mensajes como se mencionó en la sección 3.1.3.3. para iniciar o detener la toma de datos. Además, esta orden se guarda en la base de datos, se exporta en formato Json en una URL, para luego poder obtener los datos por un código Javascript mediante una petición **GET** y mostrar el estado de la toma de datos al usuario en la vista. El método GET se utiliza para solicitar datos de un recurso específico.

7.3. MONITOR PISCICULTURA

En esta vista cualquier persona puede visualizar la información de los estanques pero queda abierta la posibilidad de poner restricción de usuarios a esta vista. Como podemos ver en la Fig. 25(c) en la esquina superior izquierda se puede seleccionar el estanque de interés, en la parte izquierda se pueden visualizar los últimos valores de las variables medidas y a la derecha una gráfica en tiempo real en la cual se seleccionan las gráficas que se quieren ver. Para poder mostrar estos datos fue necesario exportar en una URL en formato JSON los últimos valores de las variables, para luego ser leído por Javascript mediante una petición GET. Además, vemos en la parte inferior derecha la opción de descargar los datos, del estanque correspondiente en formato JSON.

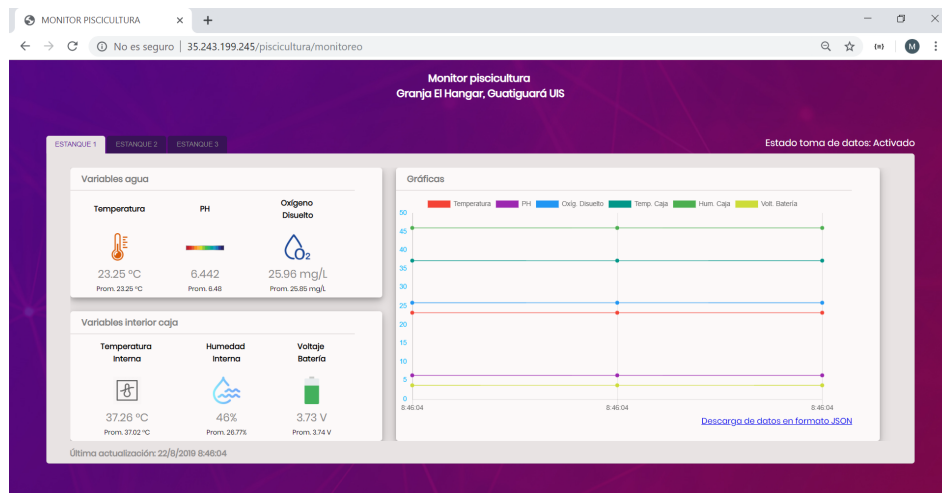
²⁷ Es un formato de texto sencillo para el intercambio de datos.

Figura 25. Vistas de la aplicación web. (a) Registro de estanques, (b) Toma de datos, (c) Monitoreo.



(a)

(b)



(c)

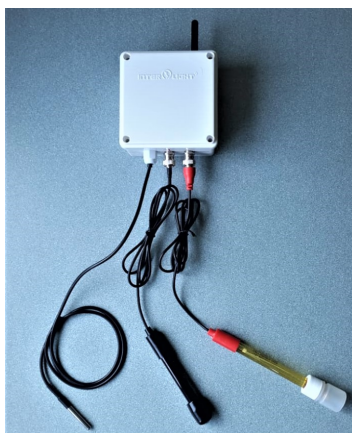
8. VALIDACIÓN DE LA COMUNICACIÓN LOCAL

En este capítulo se presenta la etapa final de la metodología para el desarrollo de soluciones IoT, que consiste en la validación por medio de pruebas de comunicación donde se evidencie el correcto funcionamiento del enlace LoRa.

8.1. MONTAJE FINAL

Para el montaje final del nodo en estanque se decidió proteger los componentes electrónicos poniéndolos dentro de una caja derivación, la cual resiste la intemperie gracias a que cuenta con un grado de protección IP67. Para que las sondas que van en el agua del estanque se puedan conectar con la electrónica dentro de la caja, fue necesario abrir unos agujeros a la medida en la parte inferior; el mismo procedimiento fue necesario en la parte superior para la antena. En la Fig. 26 podemos observar el nodo en estanque terminado.

Figura 26. Modelo electrónico funcional final del nodo en estanque.



8.2. PRUEBAS DE COMUNICACIÓN

Se realizaron pruebas en la comunicación para verificar que lo que se envía desde el nodo en estanque, corresponde con lo que se recibe en el nodo central. Estas se realizaron en los estanques piscícolas de la granja UIS El Hanger en Guatiguará, lugar donde funcionará el sistema de monitoreo. Como se cuenta con un solo nodo en estanque, este se ubicó en la posición de cada uno de los 5 estanques que tiene la granja El Hanger como se muestra en los puntos rojos de la Fig. 27, con el fin de verificar el funcionamiento en cada punto donde se replicaría el nodo. El nodo central se ubicó en el punto verde dentro de una caseta y el nodo en estanque se instaló como se muestra en la Fig. 28.

Figura 27. Planos de los estanques piscícolas en la granja de la UIS *El Hanger* en Guatiguará. Nodos en estanques (puntos rojos), nodo central (punto verde).

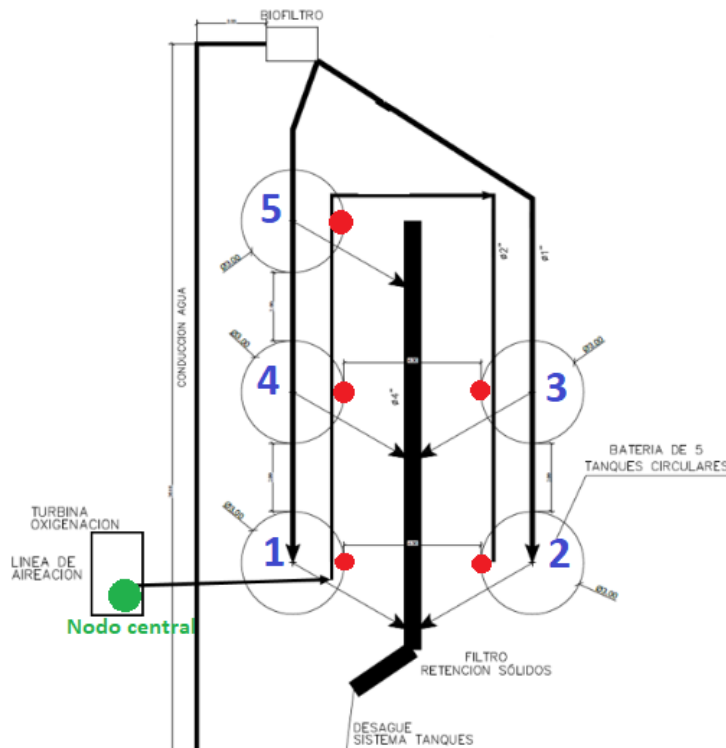


Figura 28. Nodo en estanque en la granja UIS *El Hangar* en Guatiguará.



8.2.1. Primera prueba Para esta prueba de evaluación del enlace LoRa se configuró el nodo en estanque para que enviara un mensaje cada 8 segundos, y se comparó el archivo CSV que guarda la trama de datos explicada en la sección 4.2.1. en el nodo en estanque antes de transmitir, con el archivo CSV que se guarda en el nodo central luego de recibir la información. Al realizar esta comparación, se encontró que los mensajes recibidos por el nodo central, estaban correctamente sin ninguna alteración en alguna cifra, sin embargo, se descubrió la pérdida de algunos mensajes completos, ya que en algunos casos habían más registros en el nodo en estanque en comparación con los registros del archivo .CSV del nodo central.

En la Tabla 13 se muestran los resultados obtenidos en las pruebas, allí podemos ver que se recibieron el 100 % de los paquetes en el estanque 1, el cual cuenta con línea de vista directa al lugar donde se encontraba la antena del nodo central. Además, se evidencia que en el peor caso, al menos el 95 % de los mensajes lle-

ga correctamente al nodo central, esta cifra es buena si se toman datos con una frecuencia alta debido a que no es tan imprescindible que todos los mensajes se reciban porque el valor de las variables del agua no cambian con mucha rapidez. Además se midió el RSSI (*Received Signal Strength Indicator* el cual mide el nivel de potencia de las señales recibidas en este caso, por el nodo el nodo en estanque. Podemos apreciar que los valores de RSSI más altos se obtuvieron para los estanques 1, 4 y 5 los cuales tenían una línea de vista mejor en comparación con los estanques 2 y 3, sin embargo, todos los valores de RSSI obtenidos nos indican que se recibe una intensidad buena de las señales.

Tabla 13. Resultados prueba enlace LoRa

ID Estanque	Distancia [m]	Total mensajes enviados	Mensajes recibidos correctamente	RSSI promedio [dBm]
1	3.8	100	100	-60.5
2	4.75	100	98	-67
3	6.27	100	95	-63.7
4	5.22	100	98	-69.01
5	7.12	100	97	-61.3

8.2.2. Segunda Prueba con mejoras en software Teniendo en cuenta que para tener la opción de poder tomar datos con menos frecuencia sin tener el riesgo de perder mensajes importantes, es necesario garantizar que todos los mensajes lleguen a su destino. Esto se realizó aplicando una mejora a los códigos de transmisión y recepción de los mensajes. A continuación se explica como quedo funcionando la comunicación luego de la mejora en el software:

1. El nodo en estanque(transmisor) envía un mensaje con las variables al nodo central(receptor).

2. El nodo central(receptor) cuando recibe el mensaje, calcula un número que es único en cada mensaje, por ejemplo, en este caso se calculó la suma de todos los caracteres del mensaje, luego se envía este número al nodo en estanque(transmisor).
3. El nodo en estanque(transmisor) calcula el mismo número en el mensaje que envió y lo compara con el que le llegó del nodo centra(receptor). Si ambos número son iguales significa que el receptor recibió en mensaje correctamente sin ninguna corrupción. Si los números no son iguales o si no recibió el número respuesta del nodo central(receptor), se vuelve a enviar mensaje. Se configuró para reenviar el mensaje hasta cuatro veces veces si es necesario, si luego de estas cuatro veces aun no se realiza una verificación correcta, se da por perdido en mensaje.

Luego de realizar nuevamente las pruebas, se obtuvieron los resultados de la Tabla 14, allí podemos ver que se recibieron el 100 % de los paquetes en todos los estanques, esto nos permite inferir que la mejora en el software mejoró notablemente la recepción de paquetes y permitió mejorar la confiabilidad de la comunicación.

Tabla 14. Resultados prueba 2 enlace LoRa con mejoras en software.

<i>ID Estanque</i>	Distancia [m]	Total mensajes enviados	Mensajes recibidos correctamente	RSSI pro- medio [dBm]
1	3.8	100	100	-60.8
2	4.75	100	100	-67.04
3	6.27	100	100	-64
4	5.22	100	100	-69.1
5	7.12	100	100	-61.1

9. CONCLUSIONES

- Se propuso un modelo para un sistema piscícola inteligente, que cuenta con todas las herramientas necesarias para garantizar las condiciones óptimas requeridas en el crecimiento de los peces.
- Se construyó un modelo electrónica funcional de un terminal IoT que permite el monitoreo de la temperatura, el pH y el oxígeno disuelto en el agua de un estanque piscícola. Este terminal IoT, actualmente cuenta con un nodo en estanque, un nodo central y una red LoRa, además, se caracteriza por ser escalable al poder abarcar en un futuro varios estanques simultáneamente y cuenta con respaldo de la información en una microSD ante pérdidas en la transmisión.
- Se elaboró una aplicación web en la plataforma IoT del centro TIC que permite visualizar las variables en tiempo real, registrar nuevos estanques en la base de datos y controlar la toma de datos. Además accesible desde cualquier parte del mundo mediante un dispositivo con conexión a internet.
- El sistema de monitoreo en estanques piscícolas basado en el internet de las cosas (IoT) implementado, contiene elementos en todas las capas del modelo propuesto para sistemas piscícolas inteligentes, lo cual le permite fácilmente en un futuro implementar las otras características que se plantean en este, para cumplir con las necesidades de nuevos actores.
- Las pruebas de comunicación finales del enlace LoRa en la granja *El Hangar* en Guatigurá, permitieron comprobar que se creó un canal de comunicación estable el cual confirma la llegada del mensaje correctamente. Esto nos garantiza confiabilidad en el enlace LoRa implementado.

10. TRABAJO FUTURO

Como trabajo futuro de esta investigación se plantea:

- La réplica del nodo en estanque elaborado, en los demás estanques de piscicultura de la granja *El Hangar* en Guatiguará, esto gracias a que el sistema es escalable.
- La implementación de actuadores en el sistema para controlar las variables monitoreadas. El control puede ser automático o manual desde la aplicación web.
- El modelo propuesto permite una futura implementación del terminal IoT con y sin internet. Cuando no se tenga internet el nodo central puede desplegar una red Wifi local para que los usuarios visualicen y control en el sistema a través de una interfaz gráfica que se ejecuta en un servidor local en la Raspberry. Además, la información se almacenaría en una base de datos a nivel local y cuando el nodo central cuente con internet, esta base de datos se sincronizará con la base de datos del Centro TIC para asegurar la información en la nube.
- La aplicación de técnicas de procesamiento de datos como *machine learning* para poder determinar la cantidad exacta de alimento que requieren los peces para que luego un alimentador automático en el estanque les suministre esta cantidad.

BIBLIOGRAFÍA

AUNAP. *La geomembrana, una opción para la piscicultura*. 2016 (vid. pág. 23).

BRUEGGE, B. y A. DUTOIT. *Ingeniería de software orientada a objetos*. Pearson Education, 2002 (vid. págs. 39, 46).

CAMPS PARÉ Rafael y CASTILLAS SANTILLÁN, Luis Alberto. *Bases de datos*. Universitat Oberta de Catalunya, 2005 (vid. pág. 74).

DJANGO. *Working with forms*. 2019 (vid. pág. 78).

DUY, N. T. K. y col. “Automated monitoring and control system for shrimp farms based on embedded system and wireless sensor network”. En: *In Electrical, Computer and Communication Technologies (ICECCT), 2015 IEEE International Conference on* (pp. 1-5) 18 (2013), págs. 1-5 (vid. pág. 21).

EL-KHOURI VIDARTE, Namir. *Adaptación e implementación de un sistema autónomo de ajo coste de monitorización de calidad del agua en tiempo real*. Trabajo de Grado de ingeniería de tecnologías y sistemas de telecomunicación. Madrid-España: Universidad Politécnica de Madrid, 2016 (vid. pág. 61).

ENCINAS, C. y col. “Design and implementation of a distributed IoT system for the monitoring of water quality in aquaculture”. En: *Conference: 2017 Wireless Telecommunications Symposium (WTS)* (2017) (vid. pág. 22).

FAO. *Mejora de la calidad del agua en los estanques* (vid. págs. 50, 52, 55).

— *Visión general del sector acuícola nacional Colombia*. (Vid. pág. 14).

- FUNDATION, DJANGO SOFTWARE. *Django Documentation*. 2019 (vid. pág. 76).
- GAUCHAT, Juan D. *El gran libro de HTML5, CSS3 y Javascript*. Marcombo, 2012 (vid. pág. 77).
- GOODMAN, Danny. *JavaScript Bible*. Hungry Minds, Inc., 2001 (vid. pág. 78).
- IEEE. *IEEE Std 830-1998- IEEE Recommended Practice for Software Requirements Specifications*. 2009 (vid. pág. 32).
- MASSÉ, M. *REST API Design Rulebook*. O' Reilly Media, 2012 (vid. pág. 68).
- MERINO María Claudia y BONILLA, Sara Patricia y BAGES Fernando. *Diagnóstico del estado de la Acuicultura en Colombia*. Plan Nacional de Desarrollo de la Acuicultura Sostenible en Colombia. Colombia: Ministerio de Agricultura y Desarrollo Rural, Autoridad Nacional de Pesca y Acuicultura (AUNAP), Organización de las Naciones Unidas para la Agricultura y la Alimentación (FAO), 2013 (vid. pág. 22).
- MKDOCS. *Django REST framework*. 2011 (vid. pág. 69).
- MQTT, OASIS. *MQTT Version 3.1.1*. 2014 (vid. pág. 72).
- NAVARRO, A. A., J. B. PADILLA y J. J. PRÍAS. "Construcción de un Sistema de Instrumentación para la Medición de la Temperatura, pH y Oxígeno Disuelto presentes en la Piscicultura bajo Condiciones de Estanque Artificial". En: *Scientia et Technica Año XVIII* 18 (2013) (vid. pág. 21).
- POWELL, Thomas A. *HTML: The Complete Reference*. McGraw-Hill, 1999 (vid. pág. 77).

SCIENTIFIC, ATLAS. *Dissolved Oxygen Probe*. 2019 (vid. pág. 56).

— *pH Probe*. 2019 (vid. pág. 53).

UIT. *Descripción general de Internet de los objetos*. 2012 (vid. pág. 27).

ZHU, X. y col. "A remote wireless system for water quality online monitoring in intensive fish culture". En: *El Sevier Computers and Electronics in Agriculture* (2010) (vid. pág. 20).