

## Desarrollo de una aplicación móvil para el monitoreo de parámetros en la acuicultura

Gabriel Pérez Pérez<sup>1</sup>, Miguel Angel Rojas Andrade<sup>2</sup>, Uriel Martínez Jiménez<sup>1</sup>, Magaly Ortega Roa<sup>1</sup>, Ximena B. Guzmán González<sup>1</sup>, Ileana Zamora-García<sup>3</sup>, Juan-Pablo Ramírez-Paredes<sup>3</sup>

<sup>1</sup>Licenciatura en Ingeniería Mecatrónica, División de Ingenierías Campus Irapuato-Salamanca, Universidad de Guanajuato

<sup>2</sup>Licenciatura en Ingeniería en Sistemas Computacionales, División de Ingenierías Campus Irapuato-Salamanca, Universidad de Guanajuato

<sup>3</sup>Departamento de Ingeniería Electrónica, Universidad de Guanajuato

gperezperez@ugto.mx, ma.rojasandrade@ugto.mx, u.martinezjimenez@ugto.mx, m.ortegaraoa@ugto.mx, xb.guzmangonzalez@ugto.mx, idr.zamora@ugto.mx, jpi.ramirez@ugto.mx.

### Resumen

Con el paso de los años, el incremento de la población obliga a implementar técnicas que sustenten sus necesidades básicas, por lo que el mejoramiento de ellas es un punto importante para tratar. Una de las técnicas de producción de alimentos que se proponen para hacer frente al crecimiento poblacional es la acuicultura, que conlleva muchas limitantes si no se tienen los conocimientos y la experiencia necesaria. En el presente trabajo mostramos nuestra solución para el monitoreo de un biofiltro para el tratamiento de agua de un estanque acuícola, incluyendo el diseño y la programación de una aplicación móvil para la visualización de datos relacionados al filtro. La finalidad de este trabajo es facilitar la producción de las especies acuáticas y ayudar a los usuarios a visualizar y almacenar sus datos de manera eficaz.

**Palabras clave:** acuicultura; biofiltro; aplicación móvil; Internet-de-las-Cosas

### Introducción

En la actualidad, la sustentabilidad alimentaria es un tema de gran interés, ya que estudios globales estiman que para el año 2050, la producción de alimentos debe incrementarse al menos un 70% debido al aumento de la población. Para responder a dicha necesidad, se ha optado por desarrollar diferentes procesos productivos, por ejemplo, la **acuicultura**, que es la técnica de producción de especies acuáticas en estanques de manera controlada. Un problema común en este proceso es la renovación del agua, pues la acuicultura requiere un elevado gasto del líquido, lo que genera un impacto negativo en el medio ambiente, pues es una sustancia vital para el ser humano, y hoy en día es un recurso sumamente limitado. Una posible solución combina el monitoreo de los parámetros de la calidad del agua con un biofiltro para su limpieza, como se detallará más adelante.

Para determinar una calidad de agua óptima en un estanque, se analizan varios parámetros importantes: la temperatura (rango tolerable: 25°-30°), el pH (escala de entre 7 y 9), el oxígeno disuelto (OD) (valores >5 ppm) y el amonio, este último es un compuesto proveniente de los desechos de los peces, y en concentraciones mayores a 3 ppm puede afectar el desarrollo de éstos y causar su muerte.

El factor principal que determina la proporción de amoníaco en relación con el amonio en el agua es el pH, por lo tanto, cuando éste es mayor a una escala de 9, el amonio se convertirá en amoníaco (*Reactivos y Equipo S.A de C.V, 2021*). El amonio en su forma ionizada (NH<sub>3</sub>), es tóxico cuando el pH y la temperatura son altas, y el oxígeno disuelto es bajo. A su vez, la temperatura juega un papel importante, pues su aumento o decremento afecta indirectamente en el desarrollo del animal, pudiendo llegar a la muerte, el tipo de especie que se esté tratando en el estanque y del tipo de clima de proveniencia definirá el rango de temperatura

tolerable, pero por lo general se usa el rango mencionado más arriba.

Sin embargo, existen procedimientos que ayudan a reducir la huella hídrica en la producción de peces para el consumo humano, conocidos como Sistema de Recirculación Acuícola (SRA), los cuales hacen referencia a métodos para tratar el agua para su reúso. Es por ello, que se implementa un sistema denominado *biofiltro*, el cual consiste en tener un sustrato que tenga un área superficial adecuada y suficiente para el crecimiento de bacterias nitrificantes específicamente del género Nitrosomas, es decir, que toman el amonio como su alimento y lo transformen a nitrito y nitrato, con el fin mejorar la calidad del agua. Pero, para operar un sistema de filtración biológica, se requiere de cierto grado de conocimiento, por lo que los acuicultores con poca experticia tendrán problemas y sus riesgos de pérdida de cosecha serán mayores.

Así pues, hemos diseñado e implementado una aplicación móvil desarrollada en Android Studio que le permite al usuario monitorear los parámetros de temperatura y pH. El uso de teléfono móvil es muy cotidiano hoy en día, según el INEGI, el 75.5% de la población mexicana cuenta con uno de estos dispositivos, lo que eso equivale a aproximadamente 88.5 millones de habitantes mayores de 6 años en el país (INEGI, 2022). Por esta razón, los teléfonos móviles se pueden utilizar como un medio eficaz para cubrir el cuidado diario de parámetros como los mencionados anteriormente, de la mano de la implementación del Internet de las Cosas. Se espera que el uso de esta aplicación genere un impacto positivo en la producción de peces, y que les permita a los acuicultores ahorrar tiempo y un bajo riesgo de pérdida de producto en función a los factores antes mencionados.

El estanque es monitoreado por sensores que miden las variables mencionadas en tiempo real. Dichos datos obtenidos por los sensores son enviados a una placa de desarrollo Wi-Fi ESP8266 para ser procesados y enviados a la plataforma ThingSpeak (*Crespo, 2018*), una plataforma de análisis de IoT (*"Internet of Things"*, por sus siglas en inglés), en la que se procesa la información, y se generan gráficas con estos datos, para mostrarlas desde la aplicación móvil al usuario.

## **Materiales y métodos**

### **Conexión de sensores al biofiltro**

Inicialmente nos enfocamos en obtener dos de los parámetros importantes antes ya mencionados: temperatura y pH, por lo que se utilizaron dos sensores para la medición de estos parámetros, los cuales son el sensor PH-4502C para obtener el valor de pH en un líquido, y el sensor digital DS18B20 para la temperatura del ambiente donde se encuentre.

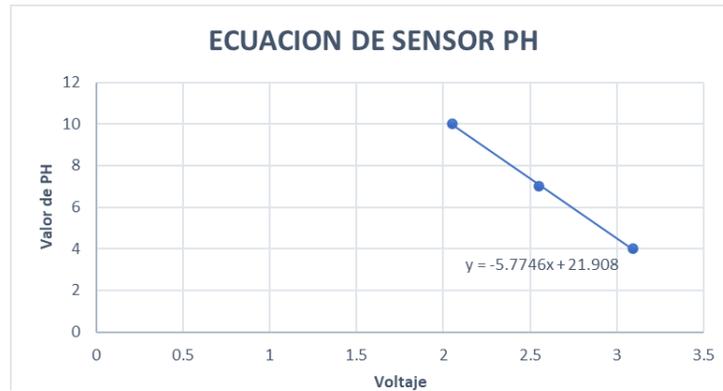


Figura 1: Ecuación lineal de voltaje vs pH

Para poder utilizar el sensor de pH es necesario calibrarlo, por lo que se requieren tres valores de pH distintos, que serán valores de referencia, ya que este dispositivo nos da un voltaje diferente dependiendo del valor de la medición, para ello se hace una calibración para un valor de pH de 4, 7 y 10, los cuales nos darán distintos voltajes; en la Figura 1 se pueden observar cómo con ayuda de estos tres valores se logra obtener una ecuación lineal que se utilizará para programar un dispositivo embebido con microcontrolador y así lograr obtener el valor de pH de cualquier líquido.

En el caso del sensor digital DS18B20, la calibración es un proceso más sencillo debido a que sólo se agrega el código que proporciona la empresa para plataformas compatibles con Arduino, y de esta manera se logra obtener el valor de temperatura del ambiente donde esté el sensor. En este caso, dentro de tanques con líquido, debido a que su estructura les permite estar dentro de los líquidos sin dañarse.

Una vez que se tienen los sensores listos para utilizar los conectamos con una tarjeta basada en el chip ESP8266. El chip ESP8266 es un dispositivo de bajo costo, con capacidad de conexión Wi-Fi incorporada, que incluye su propia implementación del protocolo TCP/IP. Es producido por la empresa Espressif Systems de China. El procesador del ESP8266 tiene una arquitectura RISC de 32 bits, con velocidad de reloj de 80 MHz. Su memoria de almacenamiento flash puede variar entre 512 KB y 4 MB, aunque soporta hasta 16 MB. La capacidad de conexión Wi-Fi de este dispositivo soporta cifrado WEP y WPA/WPA2. Cuenta además con 17 pines de entrada salida (GPIO), soporta el bus de interfase serial para periféricos (SPI por sus siglas en inglés) y tiene pines dedicados para comunicación por el estándar UART. Tiene también entrada analógica por medio de un ADC de 10 bits.

En la Figura 2 se describe cómo están conectados los sensores a la placa con ESP8266, los cuales se sumergen dentro del biofiltro para así lograr obtener los datos de los parámetros a analizar.

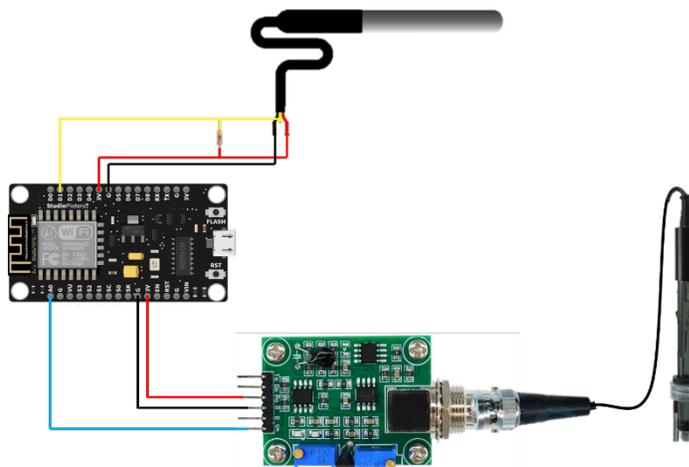


Figura 2: Diagrama de conexión de componentes

Posteriormente, después de conectar los sensores se prosigue a conectar la placa para lograr realizar el envío de nuestros parámetros mediante internet a nuestra aplicación móvil.

### Transferencia de datos

Una vez recolectados los datos gracias a los sensores que se encuentran en el biofiltro, se envían mediante el Chip Wi-Fi ESP8266 a la plataforma ThingSpeak, que se encargará de almacenar y procesar dicha información. ThingSpeak es una plataforma en línea de análisis para el Internet de las Cosas (IoT), propiedad de la empresa Mathworks. Aunque está diseñada principalmente para utilizar el software MATLAB de la misma empresa, tiene integraciones con diversos dispositivos y plataformas, y soporta los protocolos más comunes del IoT, tales como MQTT y las interfaces REST. En el caso del chip ESP8266, ThingSpeak ofrece una biblioteca de funciones específica para que los usuarios del servicio puedan habilitar fácilmente el envío de datos a la plataforma.

Para que se puedan almacenar los datos en la plataforma, se debe de crear una cuenta con la que accederemos a los diferentes canales, que son espacios donde se guardarán los datos provenientes del chip ESP8266, por lo que habrá un gráfico para cada parámetro.

### Aplicación móvil

Con el objetivo de que el usuario pueda visualizar los datos recopilados por el microcontrolador, se desarrolló una aplicación móvil para los sistemas operativos Android, debido a que este es el que cuenta con la base de usuarios más amplia alrededor del mundo.

Como entorno de desarrollo se usó Android Studio en su versión 4.2.1. con el lenguaje *Kotlin*, dicha versión del IDE (entorno de desarrollo integrado) nos proporciona las herramientas necesarias y sintaxis adecuada para funcionar con otras porciones de código ya existentes; una particularidad de desarrollar para este sistema operativo es que existe una gran variedad de componentes que deben adaptarse para funcionar en conjunto, y que diferentes versiones del propio Android que pueden estar instaladas. Por lo tanto, elegir la versión del kit de desarrollo de software o SDK es una decisión bastante importante, ya que esta define el rango de dispositivos que tienen acceso la aplicación. Con esto en mente se seleccionó como versión mínima SDK 17/ Android 4.2 (Jelly Bean) y como *target* la versión SDK 26/Android 8 (Oreo), siendo una de las más utilizadas, logrando un alcance de 12 versiones de Android.

En cuestión del diseño y programación de la aplicación, se desglosa por dos términos principales: *front end* y *back end* (García, 2022). El primero se enfoca principalmente en el diseño de la aplicación, cómo está estructurada, inclusive detalles como el uso del color, fondos, tamaños, tipografía de letra, animaciones, efectos, entre muchas cosas a considerar, con el fin de que sea agradable a la vista, pero a la vez muy clara de usar. Con respecto al segundo término, su principal función es hacer que la aplicación funcione, enfocada más en la logística para la solución al problema, es decir, la parte que procesa la entrada del frontend.

### Front end

Se diseñó un menú con desplazamiento vertical en forma de tarjetas, con la intención de asemejarse a otras aplicaciones conocidas, para así volverla más intuitiva las personas, además de poder adaptarse a todo tipo de pantallas y a futuras actualizaciones en las que existan más opciones en nuestro menú principal. Dicho menú está compuesto por los siguientes elementos o *views*: *imageButtons* (botones que aparecen en la pantalla los cuales van a interactuar con el usuario), *Texts* (texto que aparece), *AppBarLayout* (barra superior de la aplicación que muestra datos de ella como su nombre, e incluso muestra botones interactivos de apoyo, como el botón de opciones, de búsqueda, etc), y *nestedScrollView* (se utiliza cuando se necesita una vista de desplazamiento dentro de otra).

Las bibliotecas de compatibilidad que se usaron durante el desarrollo fueron las que están empaquetadas como *android.support.v4:28.0.0*, que brindan una amplia gama de clases para crear apps, desde

componentes fundamentales hasta widgets de interfaz de usuario y manejo de medios. Muchas de las clases son implementaciones compatibles con versiones anteriores que es lo que más nos interesa.

De la anterior depende la implementación de las demás librerías:

- 'com.android.support:appcompat-v7:28.0.0'
- 'com.android.support:palette-v7:28.0.0'
- 'com.android.support:design:28.0.0'
- 'android.arch.lifecycle:livedata:1.1.1'
- 'android.arch.lifecycle:viewmodel:1.1.1'
- 'com.android.support:support-v4:28.0.0'
- 'android.arch.navigation:navigation-fragment-ktx:1.0.0'
- 'android.arch.navigation:navigation-ui-ktx:1.0.0'

De las cuales, se le da énfasis a las más importantes:

- **Librería *palette-v7:28.0.0***: permite aplicar colores personalizados a los elementos visuales de la aplicación. Es de gran importancia debido a que los esquemas de color son uno de los principales componentes para un buen diseño.
- **Librería *design:28.0.0***: permite emplear diferentes estilos a los componentes de las *views*, además de habilitar elementos, opciones de diseño y el uso de otras clases de *views*, externas a las que Android Studio trae por defecto.

La clase **Drawable** es una abstracción de algo que se puede dibujar, y se usó principalmente para la creación del fondo de cada uno de los *imageButtons*, permitiendo que se extiendan según los criterios de alineación definidos y con el objetivo de adaptarse a todo tipo de pantalla en diversas situaciones específicas para estos.

Con respecto a la iconografía de la aplicación, se emplearon archivos de acceso libre de la página web: <https://www.flaticon.es/>, para posteriormente ser vectorizados en Adobe Illustrator y ser tratados como un elemento *VectorDrawable*, el cual es un gráfico vectorial definido en un archivo XML, los cuales tienen como principal ventaja es la escalabilidad de la imagen; gracias a esta característica se pueden ajustar sin perder la calidad de imagen, es decir, se cambia el tamaño del mismo archivo en función de las distintas densidades de pantalla sin afectar la calidad.

En la *Activity*, el punto de entrada que se puede cargar en cualquier momento y además permite al usuario visualizar la información proporcionada por el microcontrolador, se implementó un *TabLayout*, el cual proporciona un diseño que muestra varias páginas con pestañas, es decir, que facilita el cambio entre un contenido y otro. Este widget se diseñó para usarse con la clase *ViewPager*, que es donde llega la información directamente.

## Back End

En esta parte del código únicamente se habla de la recepción de los datos, de manera general, tomando en cuenta los componentes que resaltan en la programación de este proceso.

Para que el *TabLayout* funcione correctamente se requieren dos clases: *Kotlin* y *File Kotlin*; las dos clases *Kotlin*, se crean como subclases, las cuales fungirán como fragmentos, para poder ser usados dentro de nuestro *TabLayout*, cada uno de nuestros fragmentos, se encargara de hacer funcionar su respectivo *WebView*. Para lograr esto, se tiene que crear una función *onCreateView* con llamado de los elementos y su valor de retorno de la siguiente forma, debido a que éstos se encuentran dentro de un fragmento y no de una actividad.

Posteriormente se configura el *WebView* dándole acceso al *webChromeClient*, *webViewClient* y se le concede el permiso de acceso a Internet. Para evitar que alguna página web no muestre todo su contenido correctamente o su funcionalidad se vea limitada en la nuestra, habilitamos JavaScript para finalmente decirle al *webView* que muestre el URL que le proporcionemos, la cual se obtiene del canal creado para cada parámetro en la plataforma ThingSpeak.

Por último, en el archivo *Kotlin* que se ha creado se encuentra nuestro *fragment Manager*, que es el encargado de hacer funcionar los *fragments*, cada vez que son llamados y a su vez nombrar cada uno de los *Tabs*; Para lograr este funcionamiento, se necesita de un arreglo que almacene los nombres de los *Tabs* que se mostrar dentro del *Layout* y de tres funciones dentro de nuestra clase:

- **getItem**: Se encarga de inicializar cada *fragment* cuando este es llamado de acuerdo con la posición en que se encuentre
- **getPageTitle**: retorna el título que se la asignara al *Tab*, dependiendo de la posición en que se encuentre en el arreglo.
- **getCount**: retorna el número de *fragments* que se van a estar manejando en el *TabLayout*.

### Sistema completo

Los datos recolectados por los sensores (Temperatura y pH) son enviados a la placa con chip ESP8266 para ser procesados. Estos datos fueron mostrados al usuario mediante la aplicación, tras recibir los datos desde la placa ESP8266, que cuenta con un módulo Wi-Fi que permite además de procesar los datos recolectados por los sensores, enviarlos directamente a una plataforma IoT (ThingSpeak). Los datos almacenados en ThingSpeak son representados gráficamente para ser mostrados a través de la aplicación móvil, la cual accede a los datos a través de una dirección URL proporcionada por la plataforma. En la Figura 3 se muestra un diagrama de flujo del sistema completo.

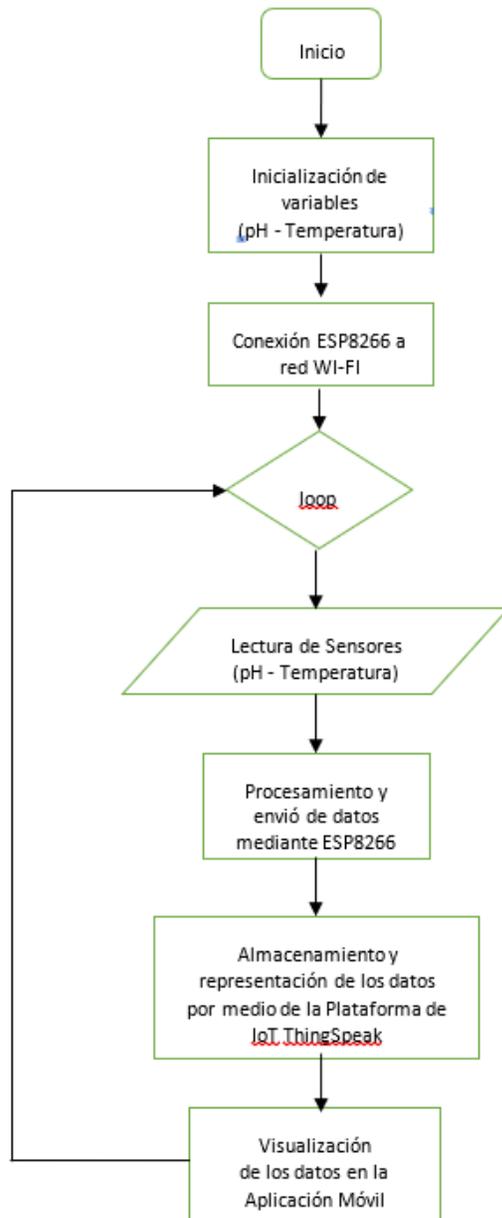


Figura 3: Diagrama de flujo del sistema completo

## Resultados

### Funcionamiento del biofiltro

A continuación, se presentan en gráficas, mediciones realizadas de manera experimental recopiladas en el periodo de un mes, con el fin de corroborar la funcionalidad del biofiltro. Se utilizaron los mismos sensores de temperatura y pH descritos en secciones anteriores, y adicionalmente se realizaron mediciones de amonio utilizando el método descrito en (Zamora-Garcia, Correa-Tome, Hernandez-Belmonte, Ayala-Ramirez, & Ramirez-Paredes, 2021). En todos los casos se obtuvieron mediciones de un biofiltro con bacterias nitrificantes, y se compararon con mediciones de otro contenedor sin bacterias.

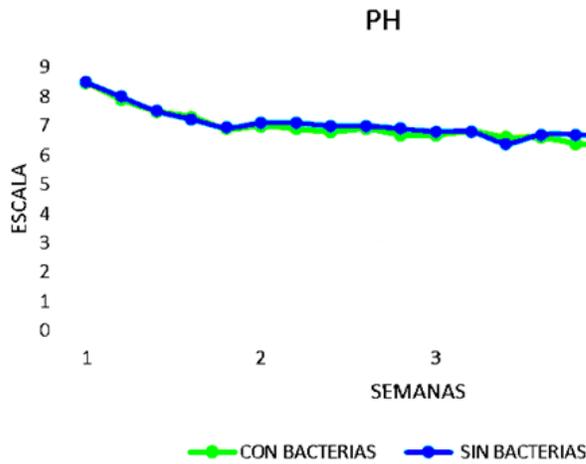


Figura 4. Modelo de comparación entre las mediciones de pH con y sin el biofiltro

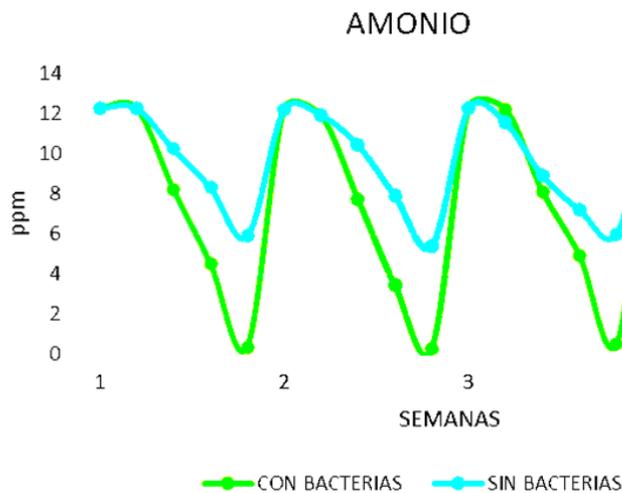


Figura 5: Modelo de comparación entre las mediciones de amonio con y sin el biofiltro

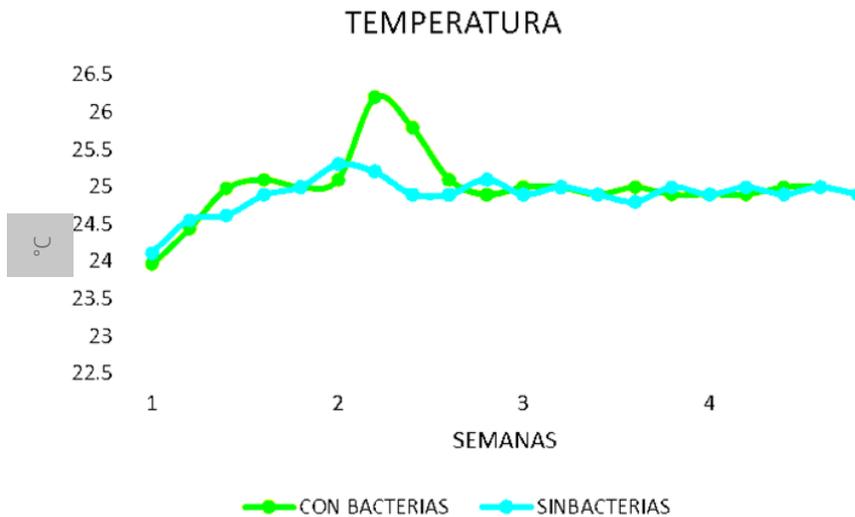


Figura 6: Modelo de comparación entre las mediciones de temperatura con y sin el biofiltro

En las figuras 4, 5 y 6 se muestran los resultados, a través de 4 semanas, de las mediciones de pH, amonio y temperatura para el biofiltro y el tanque sin bacterias. Esto permite comprobar que existe una disminución del contenido de amonio al introducir la colonia bacteriana, y que su presencia no altera significativamente los niveles de pH o la temperatura.

### Uso de la aplicación móvil

Una vez que se comprobó el funcionamiento del biofiltro, se procedió a probar la aplicación móvil desarrollada, junto con el dispositivo de adquisición de datos del Internet de las Cosas. Al principio, el usuario observa la tarjeta mostrada en la Figura 7. Para observar las mediciones, se pulsa el botón de "Monitoreo".



Figura 7: Interfaz de usuario inicial de la aplicación móvil

Cabe aclarar que los botones de “Medir Amonio” y “Alimentar” corresponden a módulos que serán desarrollados en aplicaciones futuras. El módulo “Monitoreo” le presenta al usuario la vista mostrada en la Figura 8, donde se visualizan las mediciones de temperatura y pH con respecto al tiempo.



Figura 8: Visualización de las mediciones de temperatura y pH.

## Discusión

Uno de los mayores retos para la gente que se dedica a la acuicultura es poder mantener una calidad de agua que permita la sana crianza del pescado o cualquier otra especie marina, sin embargo, no es una tarea nada sencilla, y requiere de mediciones de datos confiables y frecuentes.

En la aplicación desarrollada, la actualización de los datos es constante, por lo que mantener una bitácora desde la aplicación es práctico para los acuicultores, los cuales pueden acceder en cualquier momento del día a observar el comportamiento del estanque, y en el caso de que esté fuera del rango de tolerancia, actuar para estabilizarlo.

Si bien es un proyecto en su etapa inicial, constituye una herramienta útil para los productores acuícolas que decidan implementar un biofiltro en sus estanques. Una de las dificultades que podría conllevar el uso de esta solución es que requiere que los usuarios estén familiarizados con el uso de tecnologías móviles, sin embargo, esto puede solventarse con un breve entrenamiento. Otra limitante de la aplicación o desventaja

es que depende completamente del internet, por lo que su desconexión conllevaría a la interrupción del envío de datos. Este efecto es mitigado parcialmente con el uso de la plataforma Thingspeak como intermediaria en el manejo de datos, ya que aún si hay datos faltantes por fallos de conectividad, es posible acceder a los demás datos capturados.

## Conclusión

El principal objetivo de este proyecto es ayudar a la modernización de la acuicultura, logrando un ahorro de tiempo y delegación de tareas, gracias a la tecnología se consigue una división de la carga de trabajo, lo cual permite el dedicarse a otras labores que requieren mayor mano de obra.

Se utilizaron distintos componentes electrónicos para armar el sistema de sensores, que posteriormente enviaría la información a la plataforma a través del chip ESP8266. En la plataforma se almacenan datos, se procesan y se envían a la aplicación en forma de gráficas.

La aplicación proporciona al usuario un resumen del estado actual e histórico de su biofiltro. Para que el usuario se sienta cómodo con la aplicación, se diseñó para ser visualmente atractiva, de sencillo uso y estilo conciso, con el fin de que a aquellas personas principiantes en la tecnología no se les dificulte comprenderla.

## Referencias

- Crespo, E. (23 de Noviembre de 2018). *Thingspeak*. Obtenido de Aprendiendo Arduino: <https://aprendiendoarduino.wordpress.com/2018/11/23/thingspeak/#:~:text=ThingSpeak%20es%20un%20plataforma%20de,y%20actuar%20sobre%20los%20datos>
- García, I. J. (12 de Julio de 2022). *Backend y Frontend, ¿Qué es y cómo funcionan en la programación?* Obtenido de Servnet: <https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programaci%C3%B3n-de-una-aplicaci%C3%B3n-web>
- INEGI. (08 de Julio de 2022). *Comunicado de Prensa. Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares, 2016*. Obtenido de INEGI: [https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2021/OtrTemEcon/ENDUTIH\\_2020.pdf](https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2021/OtrTemEcon/ENDUTIH_2020.pdf)
- Reactivos y Equipo S.A de C.V. (20 de Diciembre de 2021). *Amoníaco frente a amonio: ¿cuál es la diferencia entre estas formas de nitrógeno?* Obtenido de [https://soporte.reactivosyequipos.com.mx/app/answers/answer\\_view/a\\_id/1020958/~/amon%C3%ADaco-frente-a-amonio%3A-%C2%BFcu%C3%A1l-es-la-diferencia-entre-estas-formas-de#:~:text=El%20amon%C3%ADaco%20y%20el%20amonio,fuerza%20i%C3%B3nica%20y%20la%20temperatur](https://soporte.reactivosyequipos.com.mx/app/answers/answer_view/a_id/1020958/~/amon%C3%ADaco-frente-a-amonio%3A-%C2%BFcu%C3%A1l-es-la-diferencia-entre-estas-formas-de#:~:text=El%20amon%C3%ADaco%20y%20el%20amonio,fuerza%20i%C3%B3nica%20y%20la%20temperatur)
- Zamora-García, I., Correa-Tomé, F., Hernández-Belmonte, U., Ayala-Ramírez, V., & Ramírez-Paredes, J.-P. (2021). Mobile digital colorimetry for the determination of ammonia in aquaculture applications. *Computers and Electronics in Agriculture*, 10.