TRABAJO DE GRADO
Opción Investigación o Proyecto de Grado

APLICACIÓN MÓVIL ANDROID PARA VISUALIZACIÓN DE DATOS Y RECEPCIÓN DE ALERTAS DE SISTEMAS IOT

Corporación Universitaria Remington
Facultad de Ingeniería
Ingeniería de Sistemas

Nicolas Martinez Frisneda
Carlos Hernán Ruiz Meneses

Tutor
Diego Fernando Marín Lozano

Investigación
2025

Aplicación Móvil Android Para Visualización De Datos Y Recepción De Alertas De Sistemas IoT

Nicolas Martinez Frisneda
Carlos Hernán Ruiz Meneses

Trabajo de Grado presentado como requisito para optar al título de Ingeniería en Sistemas

Tutor
Diego Fernando Marín Lozano

Ingeniería de Sistemas
Corporación Universitaria Remington
Cali
2025

Dedicatoria

El presente trabajo lo dedicamos a Dios, a nuestras familias, padres, hermanos, parejas, por ser inspiración, apoyo, ejemplo y fuente de paciencia. A ellos quienes han querido vernos salir adelante, crecer como personas y profesionales, darnos sus fuerzas para lograr superar aquellos obstáculos que se presentaron en este largo camino de aprendizajes.

Agradecimientos

Sin duda, El momento más difícil y a la vez muy importante es el recorrido en este semillero de investigación ha sido un proceso de aprendizaje, crecimiento y colaboración. Por ello, queremos dedicar este espacio y con unas pequeñas palabras mostrar nuestra gratitud a quienes han sido fundamentales en este recorrido.

En primer lugar, agradecemos al profesor Diego Marín, cuyo compromiso, orientación y dedicación han sido pilares fundamentales en este proceso. Su acompañamiento constante y su disposición para guiarnos en cada etapa han sido invaluables.

Asimismo, extendemos nuestro reconocimiento a nuestro compañero Nicolás Martínez, por su apoyo, trabajo en equipo y disposición en enseñar donde se aprendió mucho con su aporte y responsabilidad. Su colaboración ha sido clave para avanzar juntos en este proyecto.

Muchas gracias, nunca podremos compensar todo lo que hacen por nosotros. Este trabajo también es igual al esfuerzo nuestro. Gracias por confiar tanto en nosotros y por guiarnos para convertirnos en quien somos ahora. Por enseñarnos día tras día, por inculcarnos unos valores.

No podemos dejar atrás al pilar fundamental, nuestro compañerismo en todo este camino, donde nos ha llevado a apoyarnos día tras día, nos hemos aguantado nuestros días malos y nos hemos ayudado en todo, han sido todo estos años desde el primer momento que comenzamos a desarrollar este proyecto. Este párrafo va por nosotros.

Contenido

Lista de Figuras

Lista de Tablas

Resumen

Debido a la imparable digitalización que está aconteciendo en la actualidad, numerosos objetos y dispositivos que en el pasado no eran más que eso, la capacidad de monitorear y responder a los cambios ambientales en tiempo real se ha convertido en una necesidad. Por eso este este proyecto se realiza sobre el desarrollo de una aplicación móvil Android diseñada para la visualización de datos y la recepción de alertas en sistemas IoT.

Este concepto de sistema IoT internet de las cosas intenta proponer soluciones que hagan que la vida cotidiana de las personas sea mucho más sencilla, haciendo uso de la tecnología. Su principal innovación radica en la automatización de alertas en tiempo real, lo que posibilita la toma de decisiones rápidas y eficientes.

Este trabajo viene a plantear una solución en este campo de la agricultura o medio ambiente, donde gracias al diseño que se desarrolló la aplicación va facilitar el acceso a los datos desde cualquier dispositivo móvil, permitiendo a los usuarios supervisar de manera continua sus entornos y responder a eventos críticos, haciendo uso de tecnologías de vanguardia, como el uso de Aplicación web ThingSpeak, implementación de IoT, 1 un chip ESP8266 con 1 MB de memoria flash interna, para permitir a dispositivos con un chip conexiones de Wi-Fi. Además, se ponen en marcha servicios en la nube donde toda esa información que recolecta se enviara en tiempo real a la aplicación diseñada donde mantiene un sistema seguridad confiable y facilidad de uso al sistema.

Palabras clave: Internet de las Cosas (IoT), aplicación móvil, automatización, agricultura, digitalización, medio ambiente, ThingSpeak, ESP8266, nube

Abstract

Due to the unstoppable digitalization that is currently taking place, numerous objects and devices that in the past were just that, the ability to monitor and respond to environmental changes in real-time has become a necessity. Therefore, this project is focused on developing an Android mobile application designed for data visualization and receiving alerts in IoT systems.

This concept of IoT (Internet of Things) aims to propose solutions that make people's daily lives much simpler by using technology. Its main innovation lies in the automation of real-time alerts, enabling quick and efficient decision-making.

This work proposes a solution in the field of agriculture or the environment, where the design of the application will facilitate access to data from any mobile device, allowing users to continuously monitor their surroundings and respond to critical events. It utilizes cutting-edge technologies such as the ThingSpeak web application, IoT implementation, and an ESP8266 chip with 1 MB of internal flash memory to enable Wi-Fi connections. Additionally, cloud services are deployed where all the collected information will be sent in real-time to the designed application, maintaining a reliable security system and ease of use.

Keywords: Internet of Things (IoT), mobile aplication, automation, agricultura, digitalization, environment, ThingSpeak, ESP8266, cloud.

Introducción

El informe (GSMA, 2022) indica que, en el 2021, el 41% de la población mundial tenía acceso a dispositivos móviles, que la adopción 5G es un hecho, y que la tecnología móvil a contribuido al desarrollo de las economías en todo el mundo. Los smartphones están en todas partes y como tal son una herramienta perfecta para hacer llegar información directa a los interesados en forma ágil y oportuna.

Siguiendo la línea del informe de GSMA del 2022, los datos más recientes del 2023 reflejan un avance significativo en la adopción de tecnología móvil. Según el informe de Global Mobile Trends 2023 de GSMA Intelligence, se destaca que la propiedad de smartphones ha alcanzado al 55% de la población mundial, aproximadamente 4.3 mil millones de personas. Además, de los 4.6 mil millones de usuarios de internet móvil, 4 mil millones acceden a internet a través de smartphones.

Con estos avances, nuestro proyecto en curso tiene la oportunidad de impactar aún más en la sociedad, aprovechando la creciente penetración de los smartphones y las nuevas capacidades que la tecnología 5G y las tendencias emergentes ofrecen. Es un momento emocionante para fortalecer nuestras iniciativas y asegurar que la información llegue de manera eficiente y efectiva a cada vez más personas en todo el mundo.

Es por esto que, el propósito del proyecto es generar por medio de la nueva era digital una innovación importante que se pueda llevar a cabo en los dispositivos móviles para automatizar y mejorar el desempeño de aquellos procesos que tienen que depender de los cambios del clima y que de por sí, es imposible controlar. Estos procesos ya sea de cultivos o de otros sectores, se dificultaban manejar o se usaban otras prácticas para

prevenir y controlar debido a que, son situaciones impredecibles y a veces de diferente intensidad. El Internet de las Cosas (IoT) y los dispositivos móviles nos presentan esa solución y nos ayuda a facilitar el monitoreo del clima en tiempo real y en cualquier parte del mundo proporcionando herramientas que permitan mejores resultados y con la capacidad de mostrar el comportamiento del clima y de esa forma no solo impactar positivamente en la industria si no, también tomar decisiones mucho más rápidas y precisas.

Planteamiento Del Problema

¿Como se podría desarrollar, diseñar e implementar una aplicación móvil para Android que genere alertas al usuario y permita visualizar los datos en pantalla proporcionados por variables ambientales provenientes de sensores conectados a sistemas IoT?

Por negocios, entretenimiento o precaución, los dispositivos móviles son más que adecuados para entregar las alertas a los interesados. Si se dispone de una red de sensores para detectar inundaciones, un mensaje SMS o una alerta en celular es la forma más inmediata de informar a la mayoría de las personas en el área de riesgo sobre el evento. Se requiere que estos sistemas de monitoreo, redes de sensores, almacenamientos de datos, etc. puedan programar las alertas y hacerlas llegar a sus destinatarios en una forma efectiva. Los dispositivos IoT generan grandes volúmenes de datos, que pasan desapercibidos para la mayoría de las personas, pero no es útil disponer de un sistema de monitoreo sin un mecanismo apropiado para informar los eventos de interés.

En la actualidad con la entrada de nuevas tecnologías como el IoT donde una multitud de dispositivos electrónicos, desde celulares, donde todos estos dispositivos entran a la vida de las personas en forma de wearables, con una gran ventaja que es permitir al usuario un control y manejo de forma remota desde cualquier parte del mundo, el propósito de la aplicación móvil es que se puede estar atento cualquier novedad que pase ya sea en ambiente laboral, problemas climáticos entre otros

Si bien es un hecho, que los sistemas IoT son una gran herramienta, hay que destacar que la incursión en la medición de datos ambientales desde dispositivos móviles no es una práctica muy común debido a que hay una carencia de aplicaciones móviles que permitan estas funcionalidades en tiempo real y que, por lo tanto, se mantienen aquellas prácticas que no facilitan y que, además, se tarda más en la consecución del propósito.

Justificación

En un contexto actual, el hecho de poder hacer seguimiento a variables ambientales para contribuir a la productividad de un lugar es una oportunidad para mejorar y crecer exponencialmente en la industria que se esté implementando.

Así mismo, para diferentes industrias y sectores es esencial tener control de estas variables, puesto que, les permite tener mayor capacidad de control y decisión sobre el área que se esté monitoreando, ya sea en agricultura o alguna gestión de recursos naturales donde sea útil medir el ambiente.

El proyecto contempla la posibilidad de medir y hacer seguimientos de estos datos medioambientales, por medio de la visualización de la información de una forma ordenada y precisa y además, alertas que busquen la reacción oportuna a una tendencia inusual y con ello prevenir y tomar medidas agiles y acordes

La aplicación móvil ayudara a detectar y controlar en tiempo real cualquier comportamiento no normal, permitiendo resolver estas complicaciones antes de tiempo y con datos precisos para una mayor y mejor respuesta.

Pero los interesados (dueños, intermediarios o beneficiarios de esos datos) deben disponer de mecanismos seguros e inmediatos para conocer los cambios en los datos que representen alguna situación particular. La caída o aumento repentino en un precio, la temperatura de una nevera por fuera de los limites adecuados, el bloqueo de una máquina en una fábrica, etc. sea cual sea el dato de interés, cualquier cambio fuera de lo regular debería generar una alerta que llegue de inmediato al responsable de tomar las acciones necesarias.

Su importancia reside en aspectos como la accesibilidad de los datos, ya que, podrán estar disponibles en cualquier momento y lugar, facilitando su acceso oportuno. También, proporciona un aspecto crucial como son las alertas en tiempo real, ya que, gracias a esto, el usuario tiene la posibilidad de elaborar respuestas más rápidas frente a la situación que se presente. Por último, la aplicación móvil, contribuirá a una gestión más sostenible como consecuencia de lograr una mayor capacidad de monitoreo y respuesta hacia las condiciones medioambientales.

La aplicación móvil, es la oportunidad que tienen los usuarios de tener en sus manos la facilidad de tomar decisiones, hacer seguimientos y entender el comportamiento desde un dispositivo móvil con el único fin de elevar el rendimiento y conseguir mayores beneficios.

Objetivos

Objetivo General

- Construir un sistema basado en tecnología móvil que permita la entrega de alertas y el acceso a la información relevante para datos recopilados de dispositivos IoT.

Objetivos Específicos

- Definir los requerimientos y restricciones que debe cumplir la aplicación móvil para la entrega de alertas y visualización de datos.
- Diseñar la aplicación móvil para que sea de intuitiva y de fácil uso, y que acceda de datos almacenados en la nube.
- Construir y programar dicha aplicación móvil para sistemas Android.
- Realizar pruebas de la aplicación móvil que permitan garantizar el funcionamiento del sistema en condiciones reales.

Marco De Referencia

El Internet de las Cosas (IoT) ha revolucionado la forma en que interactuamos con nuestro entorno. Antes, muchos procesos parecían inalcanzables para la automatización, pero en la actualidad, la conectividad entre dispositivos nos permite optimizar tanto tareas empresariales como actividades cotidianas. En este contexto, los procesos meteorológicos desempeñan un papel crucial, especialmente en sectores como la agricultura.

La medición precisa de variables medioambientales, como la temperatura, la humedad, la presión atmosférica y la radiación solar, proporciona un mayor control y sostenibilidad. Gracias a esta información se puede influir en la toma de decisiones agrícolas: desde la planificación de cultivos hasta la protección de cosechas ante condiciones climáticas adversas. En última instancia, estas mediciones nos permiten aprovechar al máximo los recursos naturales y garantizar la productividad del lugar.

Por otro lado, también tenemos a los dispositivos móviles, los cuales se han convertido en una herramienta esencial de las personas para realizar cualquier tipo de proceso. Estos artefactos inteligentes han tenido tanta repercusión en la vida de las personas que constantemente son actualizados y mejorados a tal punto de estar equipados con múltiples funcionalidades crecientes con la capacidad de gestionar no solo la comunicación personal, sino cada detalle de la vida digital del usuario, con el fin de brindar la mejor experiencia y comodidad en todo momento. Además, para que los dispositivos móviles funcionen en plenitud, estos, poseen una gran variedad de aplicaciones móviles para todo tipo de fines y propósitos potenciando aún más la productividad del usuario. Gracias a esto, las aplicaciones móviles se han transformado en el principal medio de comunicación

de la sociedad debido a su gran versatilidad y eficiencia al momento de adquirir información importante, veraz y segura en cualquier momento y de forma actualizada.

En la actualidad la versatilidad de las aplicaciones móviles ha ayudado a mejorar la productividad de las personas ofreciendo múltiples herramientas o sistemas que solo estaban hechos para equipos de cómputos y que ahora es posible adaptarlos y aprovecharlos a la altura de la mano y en cualquier lugar.

Ahora, con el Internet de las Cosas (IoT), las aplicaciones móviles dan otro salto en la transformación digital. Con el IoT la automatización de procesos u objetos inteligentes llevado a que sean gestionados o manipulados por aplicaciones móviles permite que los dispositivos inteligentes se impulsen en la productividad no solo de las personas si no de las mismas empresas facilitando la comunicación, interacción y obtención de la información mucho más fácil, rápida y segura.

Antecedentes

Gámez López, M. D. J. (2018). Prototipo electrónico de control y monitoreo de parámetros ambientales implementando Internet de las Cosas. Revista Tecnológica; no. 11.

En este documento se tiene como idea principal resaltar las aplicaciones que la IoT permite implementar en los diferentes sectores productivos en este caso del El Salvador.

En este caso el documento hace alusión a un aplicativo en específico como lo es el monitoreo de la temperatura y la humedad desarrollado en aplicativos móviles y webs por

medio de un sistema con dispositivos, tales como sensores y actuadores, entre otros, con el fin de generar soluciones acordes con las necesidades del sector productivo, utilizando hardware y software de bajo costo que incluya la capacidad de comunicarse con otros aplicativos para generar mayor valor a los procesos de negocio. Con ese proyecto se ha contribuido a controlar y registrar parámetros ambientales, así como alertas o notificaciones oportunas a e-mail que permiten mantener salas de equipos de comunicaciones con los parámetros ambientales dentro de las tolerancias definidas por el fabricante.

Hernández Siachoque, Nicolas (2021) Diseño e implementación de un sistema IoT para monitorear calidad del aire.

Esta investigación descubre cómo se podría implementar una solución de IoT para aumentar muchas funciones en una red de calidad del aire. Características como nodos estáticos y móviles que pueden ofrecer una medición en tiempo real de la calidad del aire. El diseño presenta el desarrollo de un sistema embebido que mide tres variables que afectan la calidad del aire. El sensor se comunica a través del protocolo de comunicación Wi-Fi y el protocolo de aplicación.

Las capitales de todo el mundo enfrentan problemas de calidad del aire, uno de ellos es Bogotá Colombia, donde en 2021 la ciudad determinó políticas para monitorear y determinar los efectos de la contaminación del aire. Actualmente existe una red gubernamental de monitoreo de la calidad del aire que maneja 13 estaciones con intervalos de una hora para la medición de PM, O3 y CO.

Flores Zermeño, Francisco Javier; Cossio Franco, Edgar Gonzalo (2021) Aplicaciones, Enfoques y Tendencias del Internet de las Cosas (IoT): Revisión Sistemática de la Literatura

El artículo "Aplicaciones, Enfoques y Tendencias del Internet de las Cosas (IoT)"1 ofrece una revisión sistemática de la literatura sobre el estado actual del IoT, incluyendo su aplicación en dispositivos móviles. El estudio destaca que el IoT se entrelaza con múltiples disciplinas de la ingeniería y la ciencia, y brinda beneficios y mejoras a través de algoritmos de inteligencia artificial que se han entrelazado con la información recolectada por los sistemas IoT1.

El artículo también señala que la industria mejora día a día gracias a la generación de servicios orientados a IoT, los cuales son de gran utilidad. Además, tecnologías como Big Data, Inteligencia Artificial, Blockchain, Cloud y Edge Computing, Realidad Aumentada y 5G están potenciando el Internet de las Cosas, abriendo nuevas posibilidades para aplicaciones avanzadas.

Fernández Sosa, J. F. (Abril de 2021). Utilización de dispositivos móviles como herramienta de sensado en aplicaciones de IoT.

Este proyecto presenta un prototipo de software que busca mediante un conjunto de smartphones interconectadas por redes a una infraestructura más robusta de IoT con el propósito de manipular, analizar y obtener lo datos de los sensores que están incorporados en estos dispositivos para conseguir respuestas más rápidas y efectivas sobre situaciones de emergencia, salud o seguridad entre otros. Explica que, el sistema

incluye el desarrollo de un software que facilita la integración de estos teléfonos en redes de sensado a gran escala, donde cada smartphone actúa como un nodo autónomo capaz de procesar datos localmente y generar alertas contextuales en tiempo real. El propósito es maximizar la eficiencia y efectividad de las aplicaciones móviles que visualizan datos y gestionan alertas en sistemas IoT, ampliando la densidad y calidad de los datos disponibles para la toma de decisiones en los diversos escenarios. Se propone utilizar los sensores integrados en los dispositivos móviles como: GPS, acelerómetros, cámaras, micrófonos, etc., para capturar la información del comportamiento como del entorno para monitorear de forma continua y precisa

Rodríguez Aya, A. A., Figueredo Luna, J. A., & Chica García, J. A. (3 de Marzo de 2018). Sistema de control y telemetría de datos mediante una aplicación móvil en Android basado en IoT para el monitoreo de datos.

El articulo presente nos propone una aplicación móvil desarrollado con software y hardware accesible o de bajo costo para pequeñas y medianas empresas. Manifiesta que es posible diseñar una aplicación con sistemas IoT utilizando herramientas de fácil obtención para crear un sistema de control y telemetría basado en el Internet de las Cosas que permite el monitoreo y gestión remota de los dispositivos electrónicos de baja potencia. Además, presentan las herramientas a usar como: la tarjeta WeMos D1 mini, que incorpora el microcontrolador ESP8266 y que se usa para capturar y transmitir los datos y gestionar los dispositivos conectados a la red. Por otro lado, La WeMos D1 mini se comunica con la plataforma Thinger.io, que se encarga de hacer uso de la información

en la nube, permitiendo que los usuarios accedan a los datos y controlen los dispositivos desde cualquier lugar con acceso a internet. Por último, el diseño de la aplicación lo desarrollan con la herramienta AppInventor, la cual permite crear aplicaciones de forma visual y sencilla, y que facilita la interacción del usuario con el sistema. El articulo precisa, que el propósito es poder ofrecer una solución viable y asequible a la pequeñas y medianas empresas y que no sea costoso de implementar.

Marco Conceptual

Internet de las Cosas (IoT)

Es la manera de automatizar procesos por medio de dispositivos físicos que tienen conexión a internet y que pueden relacionarse entre sí. Además, se pueden conectar a sistemas de cómputo para ser programados según su necesidad. Es gracias a esto que se podrán recolectar los datos de los sensores para poder usarlos en la aplicación móvil.

Variables Medioambientales

Aquellos datos provenientes del entorno como temperatura, humedad, presión atmosférica y radiación solar. Que son medidos y capturados por sensores y que por medio de una placa conectada a internet y programada en un ordenador se envían a una API para poder usarlos en la aplicación móvil.

ThingSpeak

Es una plataforma web enfocada en sistemas IoT donde almacena y visualiza datos en tiempo real de variables proveniente de sensores. También es utilizado como API para manipular y consumir esos datos, por lo cual, será la herramienta que permitirá el intercambio de datos entre el módulo NodeMCU y la aplicación móvil.

Android

Los dispositivos móviles a diferencia de los equipos de cómputo poseen distintas características en su funcionalidad, es por eso, que sus sistemas operativos son diferentes. Android es un sistema operativo móvil muy popular utilizado en múltiples dispositivos móviles y que está basado en el núcleo de Linux. En este sistema operativo se desarrolló la aplicación móvil, ya que, es tan grande su popularidad que las herramientas para desarrollar son mucho más completas y robustas para el propósito del proyecto.

Kotlin

Uno de los Lenguajes de programación más populares en el desarrollo de aplicaciones móviles debido a su alto nivel y gran capacidad de crear aplicaciones móviles robustas y de gran calidad.

XML

Lenguaje de marcado extensible o en otras palabras lenguaje basado en etiquetas y atributos capaz de crear aplicaciones enfocado en el diseño de la interfaz gráfica. Permite diseñar la estructura y el estilo de la interfaz de la aplicación.

Android Studio

Editor de código para aplicaciones móviles basado en Kotlin y que es proporcionado de manera oficial por Google. Este IDE proporciona una interfaz robusta y amigable para desarrollar aplicaciones moviles de forma fácil, eficiente y segura, la cual brinda una gran cantidad de herramientas que ayudan y facilitan el desarrollo.

Diseño De La Aplicación

A continuación, se detallará la estructura e implementación del diseño de la aplicación, la cual, tiene como propósito monitorear en tiempo real las condiciones ambientales de un entorno especifico y facilitar la toma de decisiones teniendo en cuenta los datos recolectados por medio interfaces que permitan visualizarlos y alertas que ayuden a tener un mayor control y seguimiento.

La aplicación móvil que está diseñada para Android contempla aspectos importantes que contribuyeron al funcionamiento completo de la aplicación. Esos aspectos se dividen en 3: el componente de IoT compuesto por una placa programable con conexión a wifi

(NodeMCU); los sensores que capturan los datos; y la API donde se alojan los datos para después consumirlos.

La placa o tarjeta de desarrollo NodeMCU esp8266, es un sistema IoT con conexión a wifi programable, a la cual, se le conectan los sensores y por medio de un ordenador y el IDE de Arduino se programan para que estos recopilen los datos y después los envíe a una API. El IDE de Arduino, es un editor de código basado en lenguaje C++ donde se programa la tarjeta.

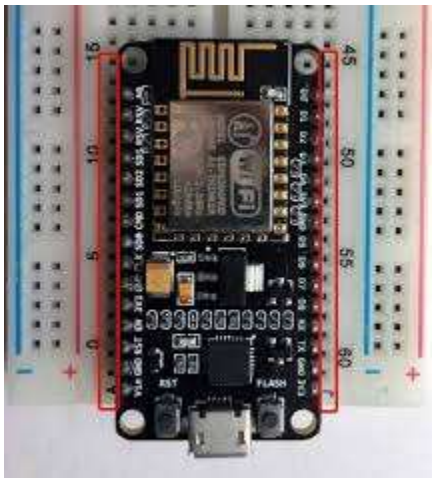

Figura 1. Modulo NodeMCU ESP8266 en una protoboard

Los sensores, son dispositivos de hardware que van conectados a la placa ya sea con ayuda de cables o soldadura. Estos sensores son los encargados de recibir o capturar los datos respectivos. Una vez conectado al módulo NodeMCU se programan a través del IDE de Arduino para que los datos sean almacenados en el mismo módulo y posteriormente sean enviados al aplicativo web. Su constitución permite capturar los datos en tiempo real y de manera constante mientras estén conectados.

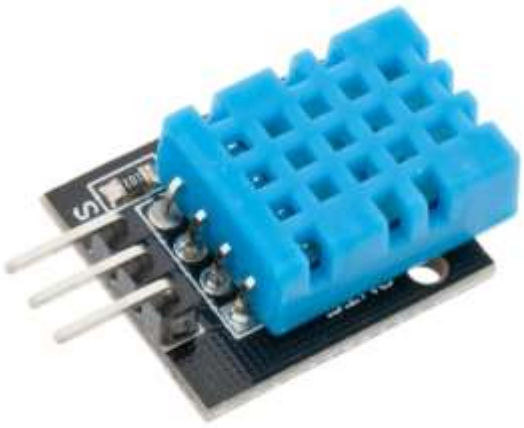

Figura 2. Sensor de Temperatura y Humedad

La API, es un puente o un intermediario que conecta dos sistemas con el fin de intercambiar o compartir datos. Para poder usar los datos se hizo uso de la plataforma ThingSpeak que es un aplicativo para visualizar datos de sistemas IoT y que también sirve como API para consumirlos. En la plataforma se estableció los campos de temperatura y humedad mientras que en el software se estableció el ID del canal y el código de la API para dirigir los datos y alojarlos en la plataforma. Una vez hecho, en la aplicación móvil se llama a la API por medio de la URL y diferentes librerias para poder hacer uso de los datos.

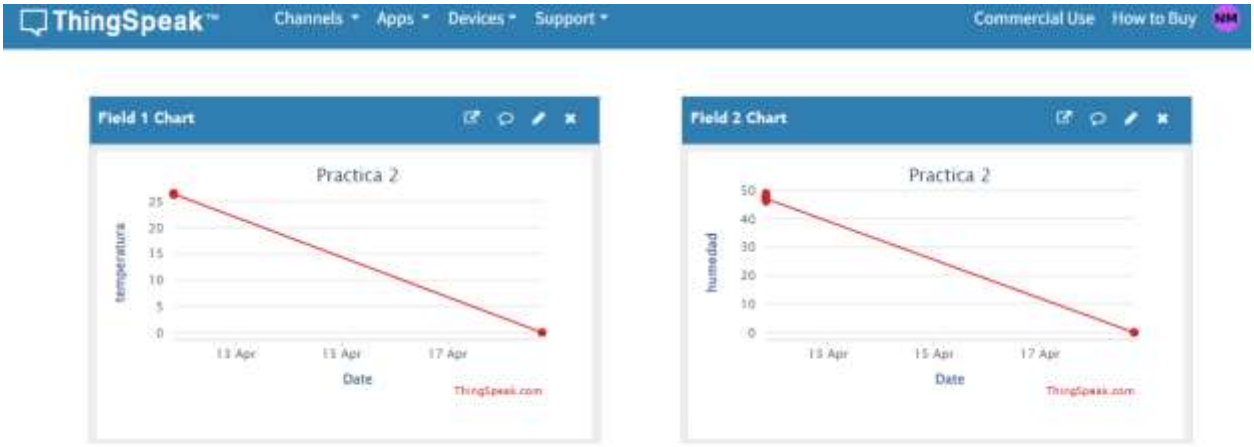

Figura 3. Grafica de temperatura y humedad en la plataforma ThingSpeak

Teniendo en cuenta lo anterior, se procede al diseño de la aplicación. Para desarrollar la aplicación móvil en Android se optó usar Kotlin con el IDE de Android Studio. Kotlin es el lenguaje de programación para aplicaciones móviles más utilizado hoy en día en el mundo gracias a su alto nivel y a su gran variedad de opciones haciéndolo robusto, completo y fácil para interpretarlo, y Android Studio es el editor oficial de Google que además permite desarrollar código de manera muy rápida gracias a la facilidad de su interfaz con el usuario. Además, también maneja lenguaje XML que es compatible con Kotlin y que se usa para los diseños de las interfaces, por lo que, con ayuda de Android Studio es posible dividir el trabajo al ubicar toda la lógica con el lenguaje Kotlin y el diseño con XML.

Para instalar el Android Studio, se debe descargar del sitio oficial de Google Developer (https://developer.android.com/studio?hl=es). Allí aparece la última versión,

luego se acepta los términos y condiciones y por último se descarga. Para instalarlo simplemente se siguen las indicaciones hasta que se instale completamente.

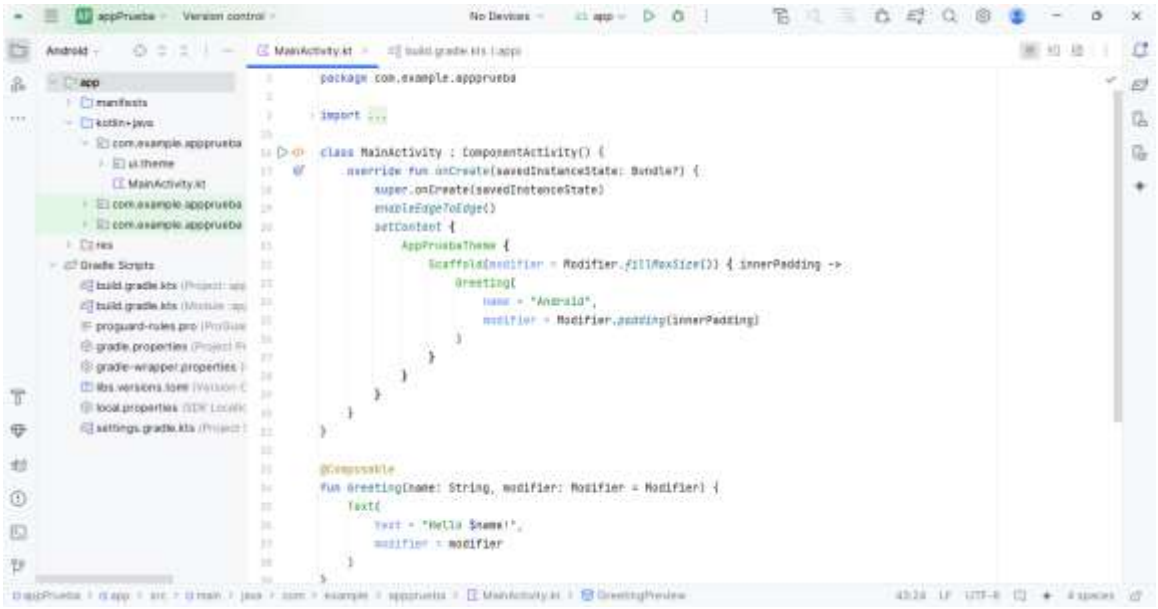

Figura 4. Interfaz del Android Studio

Una vez establecido el IDE, se inicia la estructura y desarrollo de la aplicación la cual dividimos en 4 fases. Cada fase está definida para orientar el desarrollo teniendo en cuenta la metodología usada.

Las 4 fases son:

1. Esquema o diseño de la aplicación y la integración de las dependencias.
2. Creación del menú lateral, los Fragment y la vinculación al archivo principal (MainActivity.kt).

3. Clases para la API, creación del Adapter y RecyclerView y Lógica del Fragment.
4. Creación del Login, diseño del icono de la aplicación y pruebas finales.

Fase 1:

Antes de empezar a desarrollar la aplicación se definió el esquema o diseño que iba a tener la aplicación para, de esa manera, saber que componentes y dependencias se iban a utilizar. Para el diseño, se optó por visualizar la información en forma de lista donde cada recuadro contenga el dato del sensor más la fecha de creación de forma descendente. Se emplea esta estructura debido a la gran cantidad de datos que se van a mostrar permitiendo una mejor lectura y manipulación de la información. Después, se definieron las librerías o dependencias que la aplicación va a ocupar.

```
// Implementaciones para la App
implementation(libs.androidx.fragment.ktx)
implementation(libs.androidx.drawerlayout)
implementation(libs.androidx.appcompat)
implementation(libs.material)
implementation(libs.retrofit)
implementation (libs.retrofit2.converter.gson)
implementation (libs.logging.interceptor)
implementation (libs.androidx.recyclerview)
// ****************************************
```

Figura 5. Dependencias en Kotlin

Cada uno de estos componentes se encarga de una parte importante en la aplicación, facilitando el proceso de desarrollo y proporcionando componentes que permitan un resultado adecuado y cómodo para el usuario. A continuación, se detalla cada una de las dependencias:

Tabla 1. Descripción de las dependencias

DEPENDENCIAS	DESCRIPCIÓN
Fragment	Se usa para crear las diferentes interfaces de la aplicación
DrawerLayout	Se utiliza para crear el menú lateral y poder navegar en las opciones
AppCompat	Permite desarrollar la aplicación compatible con versiones anteriores de Android, permitiendo usar componentes de usuario compatibles
Material	Son componentes de usuarios de la biblioteca material design, para lograr una interfaz moderna y atractiva para el usuario
Retrofit	Permite hacer las solicitudes HTTP para acceder a los datos de la API de forma rápida y segura
Retrofit Converter GSON	Permite convertir las cadenas Json en objetos Kotlin
Logging Interceptor	Complemento de Retrofit que permite registrar y depurar las peticiones y respuestas HTTP

Recyclerview	Permite mostrar la lista de datos de forma más eficiente en la aplicación

Además, antes de empezar el desarrollo de la aplicación, es necesario concederle el permiso de internet a la aplicación para que esta pueda realizar operaciones de red y de esa manera acceder a la API y traer los datos. Para hacer esto, se debe ir al archivo "AndroidManifest.xml" y poner la siguiente línea de código a la altura de la etiqueta de "manifest": *<uses-permission android:name="android.permission.INTERNET" />*. Es importante tenerlo en cuenta para evitar posibles errores en la aplicación y así poder hacer los procesos que necesiten de internet.

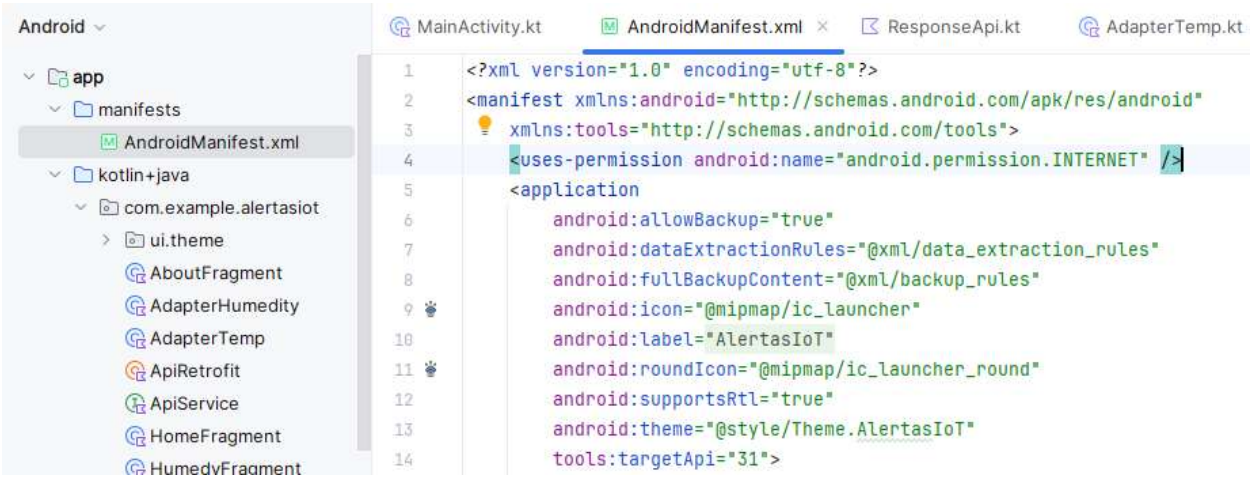

Figura 6. Configuración para el acceso a internet

Fase 2:

En esta fase se dispone a crear el menú lateral con sus opciones para después vincularlo al archivo principal (MainActivity.kt). Para ello hacemos uso del XML (Lenguaje de Marcado Extensible), este lenguaje nos permite diseñar de manera grafica la aplicación por medio de etiquetas y atributos. Además, para vincular o enlazar algún elemento o componente del XML con la lógica de Kotlin, se utiliza el ID, que son identificadores que cada componente tiene y que por medio de estos, Kotlin puede manipular el respectivo elemento

Para empezar, se crea primero un archivo XML para el menú con las opciones de inicio, temperatura, humedad, configuración, acerca de, etc. Cada opción con su respectivo icono que se crea con Android Studio y las cuales son guardadas en la carpeta “drawable”.

En otro archivo XML se crea el encabezado, la cual va a llevar los títulos. Por último se crea un archivo XML principal llamado “activity_main.xml”, donde se integra el menú y el encabezado.

Figura 7. Diseño del menú con XML

Después, en Kotlin, nos dirigimos al archivo principal del proyecto llamado “MainActivity.kt”, que es donde se conecta y se ejecuta toda la aplicación y la cual se crea automáticamente por defecto. Allí, se vincula el menú lateral con las interfaces por medio de un método llamado “onNavigationItemSelected”, el cual permite navegar entre las opciones del menú y de esa manera abrir la interfaz respectiva. Pero antes de integrar todo, se debe crear la interfaces. Para eso Kotlin, permite crear Fragments, que son pequeñas partes de la aplicación las cuales, pueden tener su propio diseño y su propia lógica, permitiendo así, utilizarlas como interfaces para cada opción. Cuando se crea un Fragment, automáticamente se crean dos archivos (el archivo Kotlin y el archivo .xml), ambos archivos se conectan automáticamente entre sí por medio de un ID. Cada

Fragment es independiente y por lo tanto su lógica y diseño no interfiere con los demás Fragments.

En este punto solo se crean mas no se implementa la lógica, se crean para vincularlas a las opciones del menú lateral desde el archivo principal y de esa manera enlazar toda el sistema.

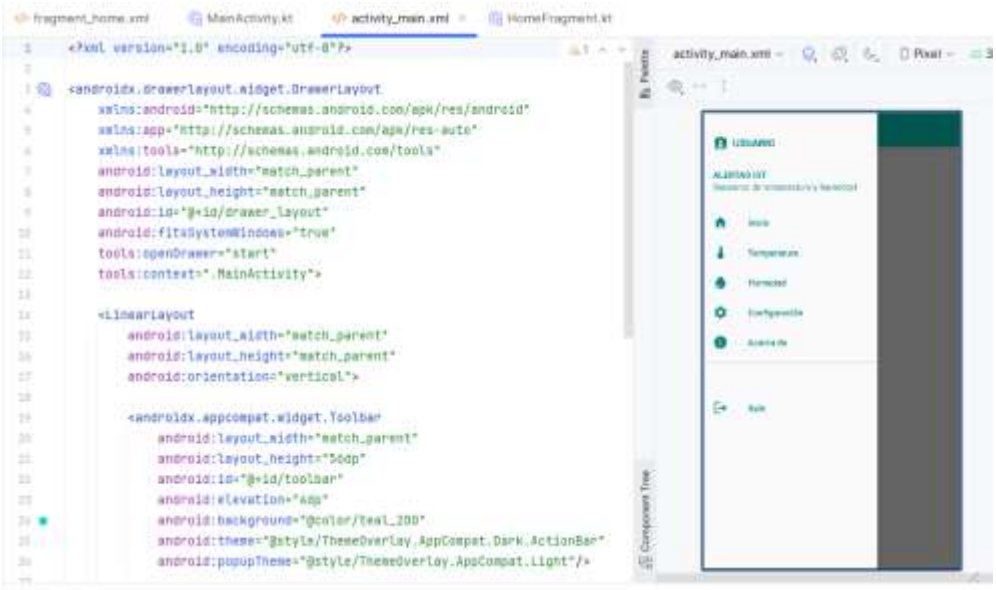

Figura 8. Código XML, el archivo activity_main.xml y el diseño del menú lateral

Fase 3:

Para esta fase se empieza a crear las clases para manejar la API y traer los datos. Para llevarlo a cabo se crearon tres clases: la primera, se encarga de hacer la solicitud HTTP para consumir los datos de la API por medio de Retrofit y el URL de la API.

MainActivity.kt ApiService.kt

```
1  package com.example.alertasiot
2
3  import retrofit2.Response
4  import retrofit2.http.GET
5
6  interface ApiService {
7      @GET("channels/2497571/feeds.json?results=15")
8      suspend fun getEntries(): Response<ResponseApi>
9  }
```

Figura 9. Clase para las solicitudes HTTP

La segunda, se crea una data-class, que es una clase donde se modelan y se almacenan los datos obtenidos por la API. Se crean cuatro variables teniendo en cuenta los campos de datos que se van extraer de la API: el entry_id que es el identificador de cada dato, field1 es el dato de temperatura, field2 es el dato de humedad y created_at es la fecha de creación del dato.

MainActivity.kt ApiService.kt ResponseApi.kt

```kotlin
package com.example.alertasiot

data class ResponseApi(
    val feeds: MutableList<Feed>,
)

data class Feed(
    val entry_id: Int,
    val field1: String,
    val field2: String,
    val created_at: String
)
```

Figura 10. DataClass para modelar los datos de la API

Y la tercera clase, es donde se hace la conversión de la cadena de datos Json a objetos Kotlin para poder manipularlos.

MainActivity.kt ApiService.kt ResponseApi.kt ApiRetrofit.kt

```kotlin
package com.example.alertasiot

import retrofit2.Retrofit
import retrofit2.converter.gson.GsonConverterFactory

object ApiRetrofit {

    private var retrofit: Retrofit? = null

    fun getInstance(): Retrofit {
        if (retrofit == null) {
            retrofit = Retrofit.Builder()
                .baseUrl("https://api.thingspeak.com/")

                .addConverterFactory(GsonConverterFactory.create())
                .build()
        }
        return retrofit!!
    }

    fun getApi(): ApiService {
        return getInstance().create(ApiService::class.java)
    }
}
```

Figura 11. Clase para la conversión de Json a Kotlin

Una vez configurada la API y los datos, se procede a implementar el RecyclerView. Para esto se debe crear primero un archivo XML donde se diseña la plantilla que llevara cada ítem de la lista, es decir, para crear el diseño de la lista basta con desarrollar una plantilla con todo el diseño de un solo ítem y de esa manera RecyclerView adaptara ese diseño en los demás ítems de la lista facilitando el desarrollo del diseño de la aplicación.

Figura 12. Plantilla para cada ítem del RecyclerView

Una vez hecho lo anterior, seguimos con la creación del RecyclerView, para ello debemos crear una clase llamada Adapter, esta clase es la que se encarga de establecer toda la configuración del RecyclerView, es decir, estructura y enlaza los datos de forma individual con la plantilla ya creada en el paso anterior y se indica la cantidad de datos a mostrar. Esta clase posee tres métodos predeterminados e importantes para organizar y crear la configuración del RecyclerView: onCreateViewHolder, onBindViewHolder y getItemCount.

Tabla 2. Métodos de la Clase Adapter

METODO	**DESCRIPCION**
onCreateViewHolder	Se encarga de crear la vista de cada elemento de la lista con la plantilla creada.
onBindViewHolder	Se encarga de asignar los datos a cada ítem de la lista.
getItemCount	Se encarga de decirle al RecyclerView cuantos elementos va a mostrar.

Tanto la interfaz de Temperatura como de Humedad tienen una Clase Adapter independiente debido a que manejan diferentes datos cada RecyclerView.

Ya establecida la API y el Adapter con la configuración del RecyclerView, nos dirigimos a implementar la lógica de la interfaz (Fragment) de la Temperatura y Humedad.

Para ello, primero vamos al XML del Fragment de temperatura y de humedad, en el creamos el componente del RecyclerView con sus atributos y su ID.

Figura 13. XML del Fragment de Temperatura con el RecyclerView

Una vez creado el componente, pasamos al Kotlin del Fragment y allí se empieza a desarrollar la lógica para mostrar los datos en la interfaz. Empezamos declarando las variables para el objeto del RecyclerView, la lista de los datos y el objeto de la clase Adapter. Seguido, se le asigna el ID del componente RecyclerView al objeto RecyclerView. Seguido, le asignamos la lista de los datos al objeto Adapter para después asignarlo al objeto del RecyclerView, para que este puede mostrar los datos. Después se crea un método para la lista de los datos, donde, se hace la solicitud HTTP a la API por

medio de las clases creadas para traer los datos y actualizar la lista del RecyclerView cada vez que haya un dato nuevo. Este método se llama justo después de que se le asigna el Adapter al RecyclerView para que los datos se actualicen cuando sea necesario.

Figura 14. Código Kotlin para el Fragment de Temperatura

Fase 4:

Esta fase se implementa el login para iniciar sesión, posterior a eso se establece el icono de la aplicación móvil y por último se hacen las pruebas de funcionamiento.

Para el login primero se crea un archivo llamado custom_edittext.xml para la configurar o personalizar los campos de texto donde se va a ingresar el usuario y la

constraseña. Además antes de ello se crea dos iconos uno de usuario y otro de candado para mejor el diseño de los campos de texto.

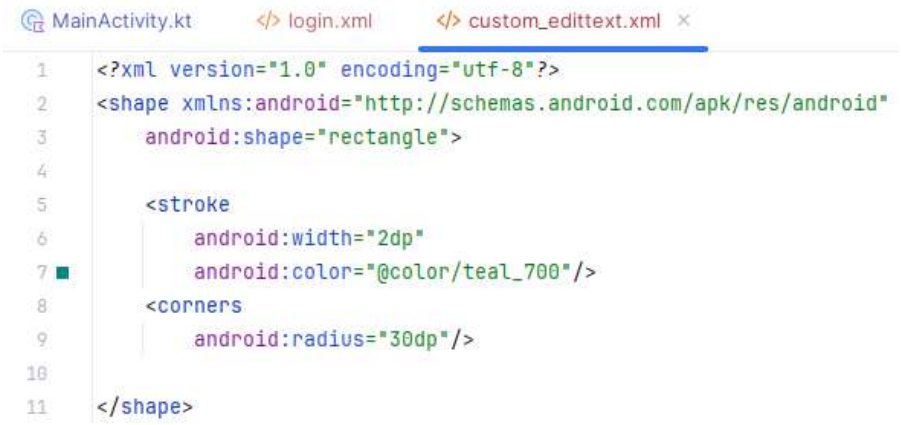

Figura 15. Archivo custom_edittext y su Código

Después se crea otro archivo llamado login.xml y allí se crea el diseño. Se crea el recuadro y dentro el título, los campos de texto, se vincula la configuración del archivo anterior para los campos y por ultimo se crea el botón.

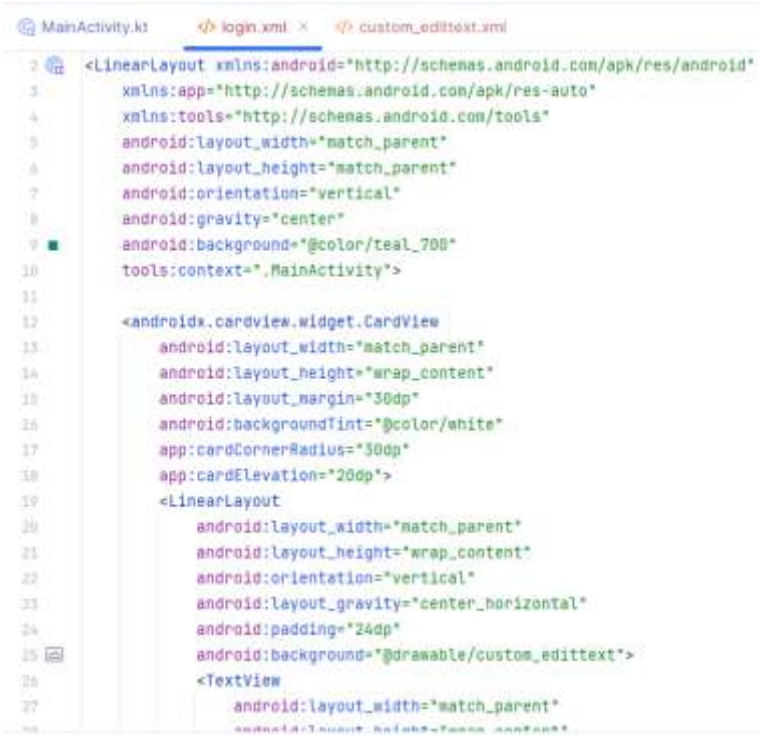

Figura 16. Diseño del login

Por último se adapta el archivo principal (MainActivity.kt), para que cuando se ejecute la aplicación se inicie en el login y una vez se ingresa las credenciales se pasa al menú principal de la aplicación. Para organizarlo, primero se crean las variables para la interfaz del login, de la aplicación, para el usuario y la contraseña, segundo se asigna el login como la primer interfaz, tercero se hace una condición para validar las credenciales y si las credenciales son correctas se ejecuta el método donde está el menú principal de la aplicación y de esa manera se vincula todo.

Figura 17. Archivo principal con el login adaptado

Para el icono, primero se seleccionó una imagen de internet relevante, se descargó en el ordenador y después con ayuda de Android Studio se configuró para que se adapte a las dimensiones de los dispositivos móviles y se vea correcto.

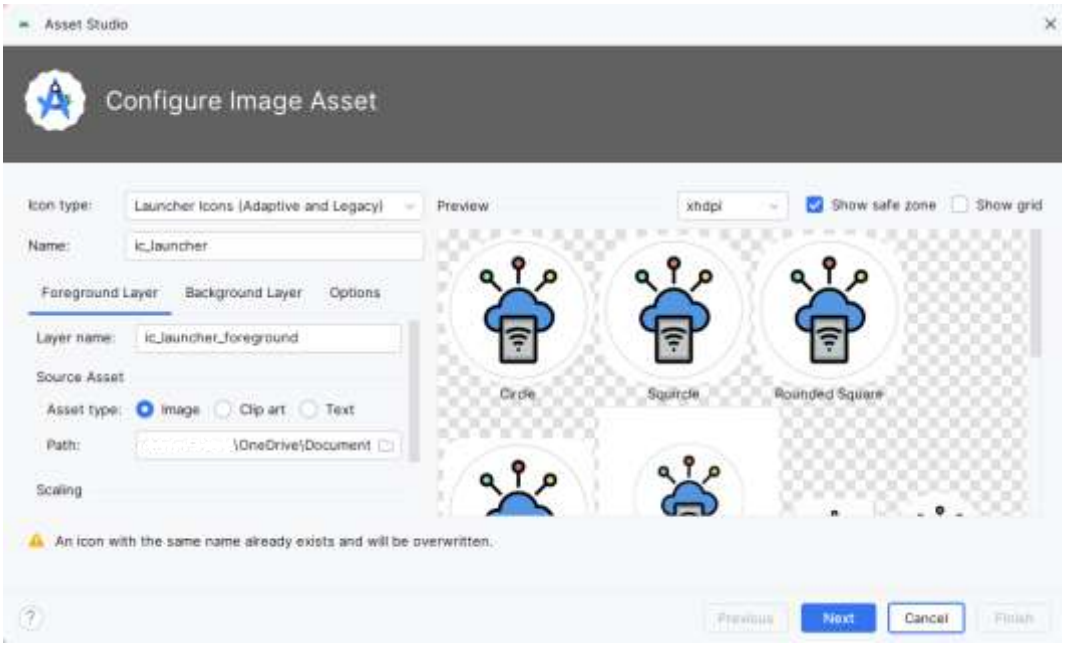

Figura 18. Configuración del icono de la aplicación

Por último, hacemos las pruebas pertinentes donde se revisa la presencia de los datos, la interacción con las interfaces, la usabilidad de la aplicación y que el diseño este acorde. Además, se revisa que no haya presencia de errores, que la aplicación no le falte ningún detalle y que el rendimiento sea optimo.

En el proceso de pruebas se corrigieron varios detalles como:

- Se estableció la lista en orden descendente para mostrar de primero el dato más reciente, esto debido a que se visualizaba los datos más antiguos lo cual no era lo esperado.

- Se limito la cantidad de datos a 15, es decir, desde la clase de la API se configuro para que solo se muestren 15 datos inicialmente, ya que, la cantidad de datos es mucha.
- Se estableció un formato de fecha adecuado para el campo de “created_at” de la API para que se visualizara de forma entendible en la aplicación mostrando la fecha y la hora ordenado y separado.
- Se organizo y redefinio el diseño de la aplicacion para darle mas estetica a las interface

Metodología

La metodología de desarrollo de software a utilizar en este proyecto es el Prototipado Rápido, esta metodología consiste en crear un prototipo preliminar el cual se evalúa y se hace retroalimentación para crear otro prototipo mejorado y de esta manera iterar las veces que sean necesarias hasta obtener el producto final. Su estrategia permite desarrollar prototipos de forma rápida y con mejora continua para un resultado perfecto.

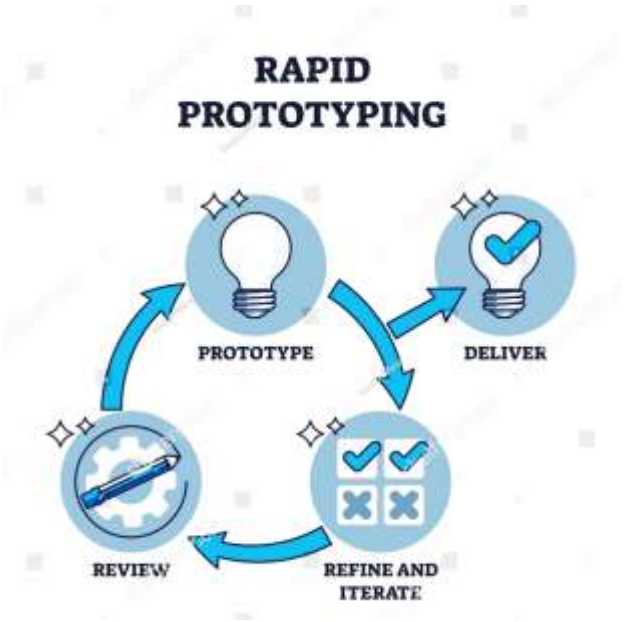

Figura 19. Definición de la Metodología de Prototipado Rápido

Para este proyecto se desarrolla un prototipo de aplicación móvil de acuerdo con los requerimientos previamente definidos donde se evalúan los resultados y se ajustan los requerimientos para el mejoramiento del nuevo prototipo. En cada iteración se corrige y se adicionan nuevas características hasta lograr el resultado deseado.

Descripción de las etapas:

Etapa 1: Se definen los requerimientos fundamentales y se contextualiza el alcance del proyecto en cuanto a sus funcionalidades.

Etapa 2: Se recopila toda la información relevante al IoT y aplicaciones móviles, se establece las herramientas para desarrollar la aplicación móvil (Android Studio y Kotlin) y se implementa los componentes relacionados y necesarios para el funcionamiento como el módulo NodeMCU ESP8266, la API y los sensores.

Etapa 3: Se elabora el primer esquema para la visualización de la aplicación y se empieza a ajustar las dependencias que se van a usar. Se crean los primeros archivos en los que se van a desarrollar (Kotlin o XML), y se diseña el menú lateral.

Etapa 4: Se empieza a crear las interfaces de temperatura y humedad (Fragments), se crean las clases para la API, se crea la clase Adapter para el RecyclerView, se crea la plantilla para el diseño de los datos y se agrega el componente RecyclerView al Fragment de temperatura y humedad.

Etapa 5: Se vincula todo en el archivo principal “MainActivity.kt”, se desarrolla la lógica de las interfaces (Fragments) y se comienza a evaluar el funcionamiento y el diseño de la aplicación.

Etapa 6: Se hace retroalimentación, se corrige funcionalidades y diseño y se itera hasta conseguir el resultado final o esperado

Análisis Y Discusión De Resultados

Este proyecto presenta como resultado una aplicación móvil intuitiva con la capacidad de visualizar datos de sistemas IoT como temperatura y humedad en tiempo real para que el usuario puede darle seguimiento a una zona y de esa manera tomar decisiones informadas que permitan mejorar la productividad y prevenir desastres. La aplicación primeramente un login para una vez se accede se un presenta un menú lateral con un encabezado que muestra el usuario y el título de la aplicación, más adelante están las opciones de Temperatura, Humedad, Configuración, Acerca de y Salir. Para las interfaces de temperatura y humedad se muestran recuadros donde se visualiza la información de Temperatura o Humedad, el día y la hora en forma descendente.

A continuación el resultado de la aplicación móvil.

En la *Figura 20*, se detalla el login con sus campos y el boton, en los campos para entrar se coloca “Usuario” y “12345”, para después darle al boton y entrar al menú principal.

Figura 20. Interfaz del Login con las credenciales

En la *Figura 21* se muestra el menú lateral con sus opciones de inicio, temperatura, humedad, configuración, acerca de y salir. Además el encabezado con el usuario y los títulos. La opción salir te devuelve al login naturalmente.

Figura 21. Menu lateral de la aplicación

En la *Figura 22* se muestra la interfaz de Temperatura y Humedad, las dos interfaces son iguales a excepción de sus datos, ya que, uno muestra los datos de temperatura el otro los de humedad y de sus colores. Estas interfaces son informativas y por lo tanto solo se va a mostrar información en orden descendente.

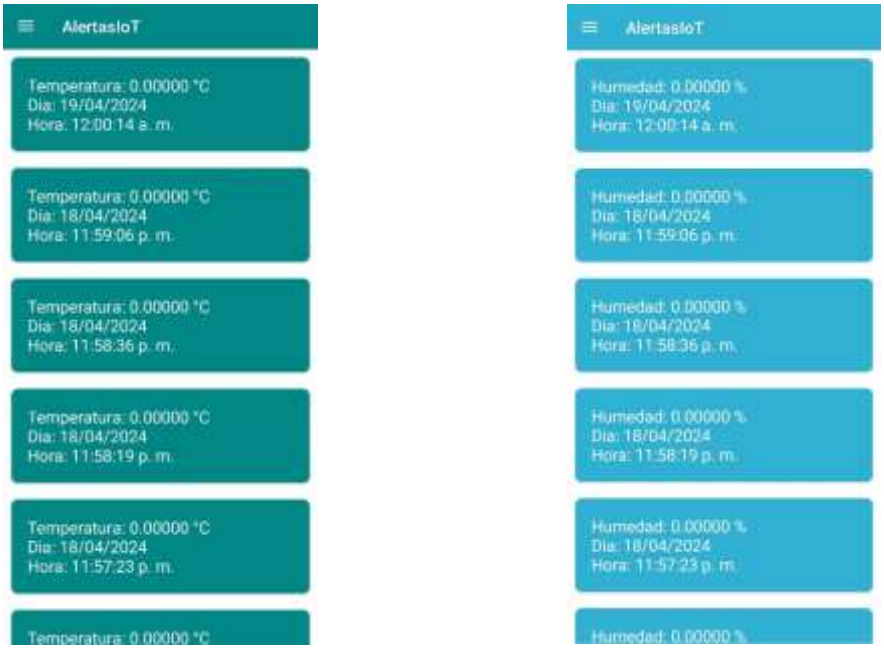

Figura 22. Interfaces de Temperatura y Humedad

Y por último la interfaz de acerca de, donde se muestra la información del proyecto y quienes lo desarrollaron.

Figura 23. Interfaz de Acerca de

Conclusiones Y Recomendaciones

El proyecto se centra en el desarrollo de una aplicación móvil, el cual se orienta hacia la monitorización efectiva de zonas donde el clima es parte fundamental mediante la integración de sistemas IoT. La aplicación permitirá visualizar en tiempo real los datos capturados por los sensores de temperatura y humedad. Este enfoque innovador no solo mejora la productividad al proporcionar información climática relevante para la toma de decisiones, sino que también actúa como una herramienta preventiva frente a desastres y optimiza el seguimiento de las condiciones ambientales en áreas específicas. Al proporcionar una solución tecnológica avanzada y accesible, este proyecto tiene el potencial de influir positivamente en la gestión de recursos y la respuesta ante emergencias, marcando un paso significativo hacia la integración de tecnologías IoT en aplicaciones móviles con fines prácticos y de seguridad.

Se recomienda el seguimiento constante a los sensores y al módulo de IoT (NodeMCU), para conservar su utilidad y calidad de su funcionamiento así como de los datos recolectados. Además, una conexión estable a internet para que los datos se mantengan actualizados constantemente y no se pierda el seguimiento dado.

Referencias

Acosta, E. J., Leon, Y. A., & Sanafria, M. W. (Marzo de 2022). *Las aplicaciones moviles y su impacto en la sociedad.* Obtenido de http://scielo.sld.cu/pdf/rus/v14n2/2218-3620-rus-14-02-237.pdf

Cardenas, G. I., & Caceres, M. M. (Enero de 2019). *Las generaciones digitales y las aplicaciones moviles como refuerzo educativo.* Obtenido de https://www.redalyc.org/pdf/7217/721778098005.pdf

Fernandez, S. J. (Abril de 2021). *Utilizacion de dispositivos moviles como herramienta de sensado en aplicaciones de IoT.* Obtenido de https://sedici.unlp.edu.ar/bitstream/handle/10915/121975/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y

Flores, Z. F., & Cossio, F. E. (Octubre de 2021). *Aplicaciones, enfoques y tendencias del internet de las cosas (IoT): Revision sistematica de la literatura.* Obtenido de https://ciateq.repositorioinstitucional.mx/jspui/bitstream/1020/543/1/Aplicaciones%20enfoques%20y%20tendencias%20del%20IoT.pdf

Google. (s.f.). *IDE de Android Studio*. Obtenido de Google Developer: https://developer.android.com/studio?hl=es

GSMA. (2023). *The Mobile Economy 2023.* Obtenido de https://www.gsma.com/solutions-and-impact/connectivity-for-good/mobile-economy/wp-content/uploads/2023/03/270223-The-Mobile-Economy-2023.pdf

Hernandez, S. N. (14 de Diciembre de 2021). *Diseño e implementacion de un sistema IoT para monitorear la calidad del aire.* Obtenido de https://repositorio.uniandes.edu.co/flip/?pdf=/bitstreams/44615508-0753-48fa-8a5f-bd6bc7d9d365/download

Rodriguez, A. A., Figueredo, L. J., & Chica, G. J. (Marzo de 2018). *Sistema de control y telemetria de datos mediante una aplicacion movil en Android basado en IoT para el monitoreo de datos.* Obtenido de https://www.revistaespacios.com/a18v39n22/a18v39n22p30.pdf

Sosa, G. S., Sosa, S. E., Gomez, D. R., Perez, V. M., & Castillo, R. F. (2020). *Impacto del internet de las cosas (IoT) en el desarrollo de aplicaciones moviles.* Obtenido de https://iydt.wordpress.com/wp-content/uploads/2020/12/3_17_impacto-del-internet-de-las-cosas-iot-en-el-desarrollo-de-aplicaciones-moviles.pdf

ThingSpeak, documentacion para la API. (s.f.). Obtenido de MathWorks: https://www.mathworks.com/help/thingspeak/readdata.html

ThingsSpeak, graficas y datos de temperatura y humedad. (s.f.). Obtenido de MathWorks: https://thingspeak.mathworks.com/channels/2497571

Anexos

Se subió todo el código completo de la aplicación móvil a la plataforma de GitHub:

https://github.com/NicolasMFDev/AlertasIoT.git

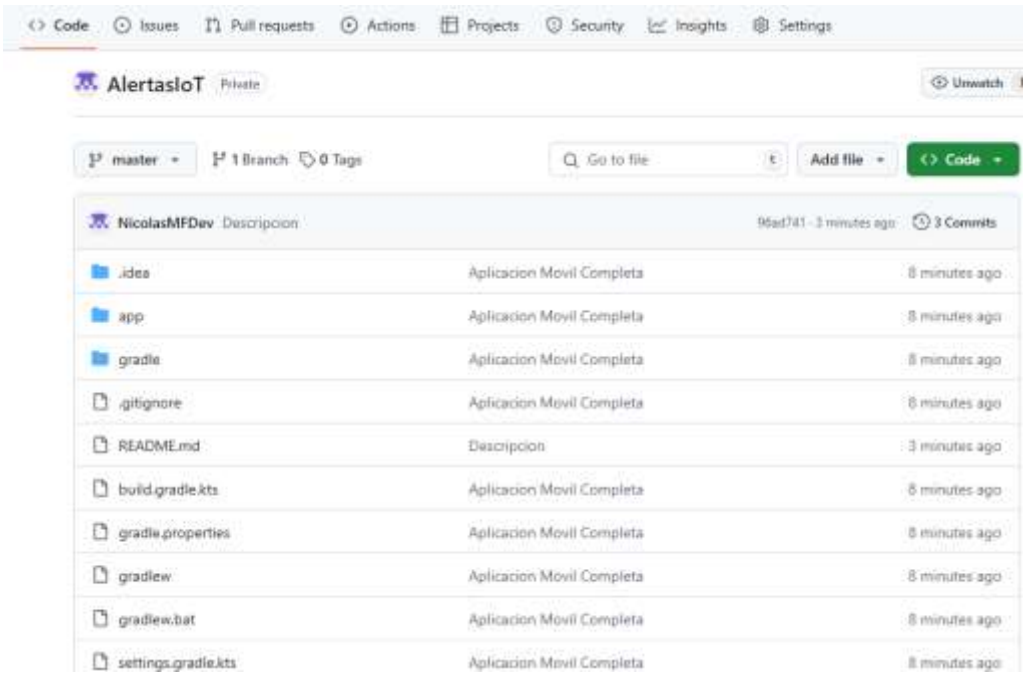

Figura 24. Código completo en la plataforma de GitHub