



Universidad Austral de Chile

Facultad de Ciencias de la Ingeniería
Escuela de Ingeniería Civil en Informática

APLICACIÓN MÓVIL PARA LA DIVULGACIÓN DEL REINO FUNGI Y DESARROLLO DEL COMENSALISMO HUMANO- HONGO

Proyecto para optar al título de
Ingeniero Civil en Informática

PROFESOR PATROCINANTE:
DR. CRISTIAN OLIVARES-RODRÍGUEZ
INGENIERO CIVIL INFORMÁTICO
MASTER EN VISIÓN POR COMPUTADORA E INTELIGENCIA ARTIFICIAL
DOCTOR EN INGENIERÍA

PROFESOR CO-PATROCINANTE
SR. JORGE MORALES VILUGRÓN
INGENIERO EN ELECTRICIDAD
DIPLOMADO EN CS. DE LA INGENIERÍA
MAGISTER EN ADMINISTRACIÓN DE EMPRESAS

PROFESOR INFORMANTE
DR. MATHIEU VERNIER
LICENCIADO EN CIENCIAS Y TECNOLOGÍAS
MASTER EN CS. Y TECNOLOGÍA
DOCTOR EN INFORMÁTICA

DIEGO ARMANDO MORA MUTIZABAL

VALDIVIA – CHILE
2023

AGRADECIMIENTOS

Quiero expresar mi más profundo agradecimiento a todas las personas que han contribuido de manera significativa en la realización de este trabajo.

En primer lugar, quiero agradecer a mis profesores guías, Cristian Olivares y Jorge Morales, por su valioso tiempo, dedicación y orientación a lo largo de todo el proceso de esta tesis. Sus conocimientos y sabias sugerencias fueron fundamentales para el desarrollo de este trabajo, y su apoyo constante me motivó a superar los retos que se presentaron en el camino. De igual forma, agradecer a la Dra. Erika Briceño por su tiempo y aportes significativos al desarrollo de este trabajo.

También quiero expresar mi gratitud a mi familia, quienes han sido mi mayor fuente de apoyo y aliento. A mi padre, Hugo Mora y a mi hermana, Vanessa Mora por su amor incondicional, comprensión y por creer en mí en todo momento.

Agradezco a todos mis amigos y seres queridos, quienes han estado a mi lado durante esta etapa de mi vida académica, brindándome su amistad y apoyo emocional.

DEDICATORIA

Dedico este trabajo a la memoria de mi amada madre, Onorinda Mutizabal. Su amor, apoyo y sabiduría fueron una fuente inagotable de inspiración en cada paso de mi vida y en la realización de esta tesis. A pesar de su partida, su legado de fortaleza y perseverancia siempre vivirá en mi corazón. Gracias por ser mi guía y mi mayor motivación.

ÍNDICE

ÍNDICE	I
ÍNDICE DE TABLAS	III
ÍNDICE DE FIGURAS.....	IV
RESUMEN.....	VI
ABSTRACT	VII
1. INTRODUCCIÓN	1
1.1 Definición del problema y oportunidad	1
1.2 Objetivos	2
2. MARCO TEÓRICO.....	3
2.1 Introducción.....	3
2.2 Revisión Sistemática.....	3
2.2.1 Desarrollo de Revisión Sistemática	3
2.2.2 Método de revisión	4
2.2.3 Resultados	6
2.1.4 Principales hallazgos.....	18
2.1.5 Conclusiones de la Revisión Sistemática.....	19
2.3 Modelo de conformación de red.....	20
2.4 Contextualización del reino Fungi.....	20
2.5 Integración con sistemas embebidos.....	22
3. METODOLOGÍA	23
3.1 Introducción.....	23
3.2 Toma de requisitos.....	23
3.3 Diseño	24
3.4 Implementación	25
4. ARQUITECTURA DE SOFTWARE.....	27
4.1 Introducción.....	27
4.2 Tecnologías y herramientas utilizadas	27
4.2.1 Android	27
4.2.2 Expo	27
4.2.3 Express JS	28
4.2.4 MariaDB.....	28
4.2.5 Docker	29
4.2.6 Ngrok	29
4.2.7 Arduino	29
4.3 Caracterización de la arquitectura	29
4.3.1 Frontend	30
4.3.2 Backend.....	30
4.3.3 Interacción Humano-Hongo.....	31
5. DESARROLLO DEL PROYECTO	33
5.1 Introducción.....	33
5.2 Resultados Toma de requisitos	33

5.2.1 Resultados entrevistas	33
5.2.2 Prototipos	37
5.2.3 Especificación de requisitos	41
Priorización de requisitos.....	41
5.3 Primer Sprint: Desarrollo de la aplicación móvil	43
5.3.1 Resumen del Sprint.....	43
5.3.2 Tareas propuestas.....	43
5.3.3 Resultados del Sprint	44
5.3 Segundo Sprint: Desarrollo del dispositivo de interacción	65
5.3.1 Resumen del Sprint.....	65
5.3.2 Tareas propuestas.....	65
5.3.3 Resultados del Sprint	66
5.4 Tercer Sprint: Integración del sistema y mejoras	75
5.3.1 Resumen del Sprint.....	75
5.3.2 Tareas propuestas.....	75
5.3.3 Resultados del Sprint	75
5.5 Validación y verificación	82
5.5.1 Prueba N°1: Envío de datos vía Bluetooth.....	82
5.5.2 Prueba N°2: Despliegue de datos en la pantalla LCD	87
5.5.3 Prueba N°3: Funcionamiento del sistema	88
6. CONCLUSIONES Y TRABAJOS FUTUROS	95
Conclusiones	95
Trabajos futuros.....	96
7. BIBLIOGRAFÍA	97

ÍNDICE DE TABLAS

Tabla	Página
Tabla 1: Criterios de selección	6
Tabla 2: Extracción de información del primer artículo	7
Tabla 3: Extracción de información del segundo artículo	8
Tabla 4: Extracción de información del tercer artículo	9
Tabla 5: Extracción de información del cuarto artículo.....	10
Tabla 6: Extracción de información del quinto artículo	11
Tabla 7: Extracción de información del sexto artículo	12
Tabla 8: Extracción de información del séptimo artículo	13
Tabla 9: Extracción de información del octavo artículo	14
Tabla 10: Extracción de información del noveno artículo	15
Tabla 11: Extracción de información del décimo artículo	16
Tabla 12: Extracción de información del onceavo artículo.....	17
Tabla 13: Extracción de información del doceavo artículo.....	18
Tabla 14: Épica: Interacción con la comunidad	36
Tabla 15: Épica: Interacción con los hongos.....	37
Tabla 16: Requisitos funcionales.....	42
Tabla 17: Requisitos no funcionales	43
Tabla 18: Tareas del primer Sprint	44
Tabla 19: Configuración del servicio MariaDB	46
Tabla 20: Configuración del servicio phpMyAdmin.....	47
Tabla 21: Descripción de endpoints relacionados a los comentarios	50
Tabla 22: Descripción de endpoints relacionados a las publicaciones	51
Tabla 23: Descripción de endpoints relacionados a los usuarios	52
Tabla 24: Descripción de endpoints relacionados a los registros Fungi.....	53
Tabla 25: Librerías utilizadas en el desarrollo	56
Tabla 26: Tareas del segundo Sprint.....	66
Tabla 27: Tareas del tercer Sprint.....	75
Tabla 28: Datos de prueba para el registro.....	89

ÍNDICE DE FIGURAS

Figura	Página
Figura 1: Esquema general del sistema.....	2
Figura 2: Selección de artículos por fuente.....	5
Figura 3: Selección de artículos por año.....	6
Figura 4: Estructura de un hongo.....	21
Figura 5: Fases del Design Thinking	23
Figura 6: Esquema de arquitectura general de arquitectura	32
Figura 7: Planificación de desarrollo	33
Figura 8: Wireflow para crear una publicación.....	38
Figura 9: Mockup para la diferenciación de usuario	38
Figura 10: Wireflow para eliminar una publicación.....	39
Figura 11: Wireflow para conectarse al dispositivo de interacción.....	40
Figura 12: Wireflow para obtener datos de los sensores	41
Figura 13: Diagrama Entidad-Relación del sistema	46
Figura 14: Organización del frontend.....	54
Figura 15: Logo de la aplicación móvil	57
Figura 16: Capturas del formulario de registro	58
Figura 17: Captura de un mensaje en la ventana informativa	59
Figura 18: Captura de la pantalla de inicio de sesión	60
Figura 19: Captura de la pantalla principal	61
Figura 20: Wireflow para crear una publicación.....	61
Figura 21: Captura de pantalla del componente de cámara	62
Figura 22: Captura de pantalla de la interfaz de reacciones y comentarios	63
Figura 23: Wireflow de eliminación de una publicación.....	64
Figura 24: Captura de pantalla del perfil de usuario.....	65
Figura 25: Placa Arduino UNO.....	66
Figura 26: Módulo Bluetooth HC-06.....	67
Figura 27: Sensor de humedad y temperatura DHT-11	67
Figura 28: Batería externa 5000mAh.....	68
Figura 29: Pantalla LCD I2C 16x2.....	68
Figura 30: Esquema de conexión del sensor DHT-11	69
Figura 31: Esquema de conexión del módulo HC-06.....	69
Figura 32: Construcción del primer circuito	71
Figura 33: Esquema de conexión de pantalla LCD I2C.....	72
Figura 34: Circuito completo del prototipo de interacción	74
Figura 35: Captura de pantalla inicial de Registros Fungi.....	76
Figura 36: Wireflow de la creación de un registro Fungi	77
Figura 37: Captura de un registro Fungi publicado	78
Figura 38: Captura de pantalla del ingreso de temperatura y humedad.....	80
Figura 39: Diagrama de proceso BPMN de la creación de un registro Fungi	81
Figura 40: Gráficos implementados con datos de prueba	81

Figura 41: Circuito Prueba N°1	83
Figura 42: Captura de la aplicación móvil utilizada en la prueba	84
Figura 43: Captura de pantalla de la conexión al dispositivo HC-06	85
Figura 44: Captura de pantalla de los datos medidos por el sensor.....	86
Figura 45: Circuito Prueba N°2.....	87
Figura 46: Despliegue de datos en la pantalla LCD I2C.....	88
Figura 47: Captura de pantalla de Registros Fungi	89
Figura 48: Captura de pantalla del formulario con los datos de prueba	90
Figura 49: Captura de pantalla de la parte de clasificación.....	91
Figura 50: Instalación del prototipo.....	92
Figura 51: Datos medidos por el sensor.....	92
Figura 52: Captura de pantalla del ingreso de los datos del sensor	93
Figura 53: Captura de pantalla del registro de prueba publicado	84

RESUMEN

Los hongos son organismos pertenecientes al reino Fungi y son primordiales para el equilibrio de muchos ecosistemas del planeta. La divulgación del reino Fungi es una tarea importante hoy en día y en este contexto surgió el proyecto interdisciplinario “Hongos y Humanos: Dispositivos para el comensalismo interespecie” que buscaba generar interés por el reino Fungi en las personas mediante la interacción entre humanos y hongos. En el marco de este proyecto, se desarrolló una propuesta con el objetivo principal de integrar lo físico y lo digital. Para ello, se ha creado un prototipo funcional de un sistema que consta de dos componentes principales.

El primer componente consiste en una aplicación móvil llamada "Micelio", desarrollada para sistemas Android utilizando el framework Expo para el frontend y una API REST desarrollada con Express JS en lado del servidor. Micelio adopta un modelo de red social. Su objetivo es fomentar una comunidad en crecimiento constante, donde los usuarios con intereses en la divulgación del reino Fungi puedan publicar sus registros y establecer interacciones entre sí. Esta aplicación proporciona una plataforma para compartir conocimientos y experiencias relacionadas con los hongos.

El otro componente del sistema corresponde a un dispositivo de interacción portátil compuesto por un microcontrolador Arduino conectado a sensores de humedad y temperatura cuyo objetivo es medir estos parámetros en un hongo. Este dispositivo funciona de manera integrada con la aplicación móvil, de forma que un usuario pueda añadir un registro de un hongo con datos relevantes para su estudio y brindando una experiencia mucho más interactiva.

Se logró que el prototipo del sistema funcione de manera integrada y cumpla con los objetivos propuestos, permitiendo el registro de observaciones de hongos desde una perspectiva innovadora y sentando así las bases para un crecimiento continuo y mejoras adicionales.

ABSTRACT

Mushrooms are organisms belonging to the Fungi kingdom and are vital for the balance of many ecosystems on the planet. The dissemination of the Fungi kingdom is an important task nowadays, and in this context, the interdisciplinary project "Hongos y Humanos: Dispositivos para el comensalismo interespecie" emerged, aiming to generate interest in the Fungi kingdom through interaction between humans and mushrooms. As part of this project, a proposal was developed with the main objective of integrating both physical and digital, resulting in a functional prototype system consisting of two main components.

The first component is a mobile application called "Micelio" developed for Android systems using the Expo framework for the frontend and a REST API built with Express JS on the server side. Micelio adopts a social network model, aiming to foster a growing community where interested users in disseminating the Fungi kingdom can share their findings and engage in interactions with each other. This application provides a platform for sharing knowledge and experiences related to mushrooms.

The other component of the system is a portable interaction device comprised of an Arduino microcontroller connected to humidity and temperature sensors, designed to measure these parameters in a mushroom. This device functions in integration with the mobile application, allowing users to add mushroom records with relevant data for further study, providing a much more interactive experience.

The prototype of the system has been successfully integrated and achieved the proposed objectives, enabling the registration of mushroom observations from an innovative perspective. It has laid the groundwork for continuous growth and further improvements.

1. INTRODUCCIÓN

1.1 Definición del problema y oportunidad

Los organismos del reino Fungi más conocidos como hongos, cumplen un rol primordial para el desarrollo y evolución de la mayoría de los ecosistemas del planeta, se encargan principalmente de la descomposición de materia orgánica y su respectiva transformación en nutrientes para otros organismos como las plantas. Asimismo, son capaces de establecer relaciones simbióticas con las plantas formando complejas estructuras llamadas micorrizas donde les proporcionan nutrientes esenciales para su supervivencia. Los hongos también son muy importantes para nosotros de formas mucho más directas como son la producción de alimentos y aplicaciones en la medicina como la producción de antibióticos o el desarrollo de medicamentos antitumorales.

A pesar de la importancia que tienen estos organismos, las personas no suelen tener conocimiento acerca de esto y, por lo tanto, no existe una consciencia colectiva de la gran biodiversidad de organismos que conforman el reino Fungi y la importancia de su conservación. La región de Los Ríos y particularmente la selva valdiviana cuenta con una gran biodiversidad de hongos y, además, reúne condiciones ideales para su proliferación, siendo la ciudad de Valdivia conocida como una “capital” Fungi en el país.

A partir de estos antecedentes surge el proyecto interdisciplinario “Hongos y Humanos: Dispositivos para el comensalismo interespecie”. Fue ideado como una iniciativa de divulgación del reino Fungi en el Jardín Botánico de la Universidad Austral de Chile en Valdivia. Una arista importante para la completitud de este proyecto es el desarrollo de una aplicación móvil con una propuesta innovadora asociada que funcione de manera integrada con estos dispositivos y que incorpore otras funcionalidades interesantes permitiendo la interacción entre humanos y hongos. Este proyecto de título se encargará del desarrollo de esta aplicación y un prototipo de estos dispositivos de interacción.

Como esquema base de este sistema, se diseñará un modelo de conformación de red o también denominado red social con el fin de promover las interacciones entre los usuarios y los hongos, estableciendo una analogía al reino Fungi que se conforma por complejos sistemas de redes denominados micelios. De manera complementaria, se integrará el uso de un microcontrolador conectado a sensores de interés con el fin de brindar una experiencia más interactiva a los usuarios (ver Figura 1) y a la vez poder capturar datos acerca de los hongos endémicos para contribuir al estudio de estos. El Jardín Botánico de la Universidad Austral de Chile será el centro de esta red y funcionará como punto de ingreso a la misma.

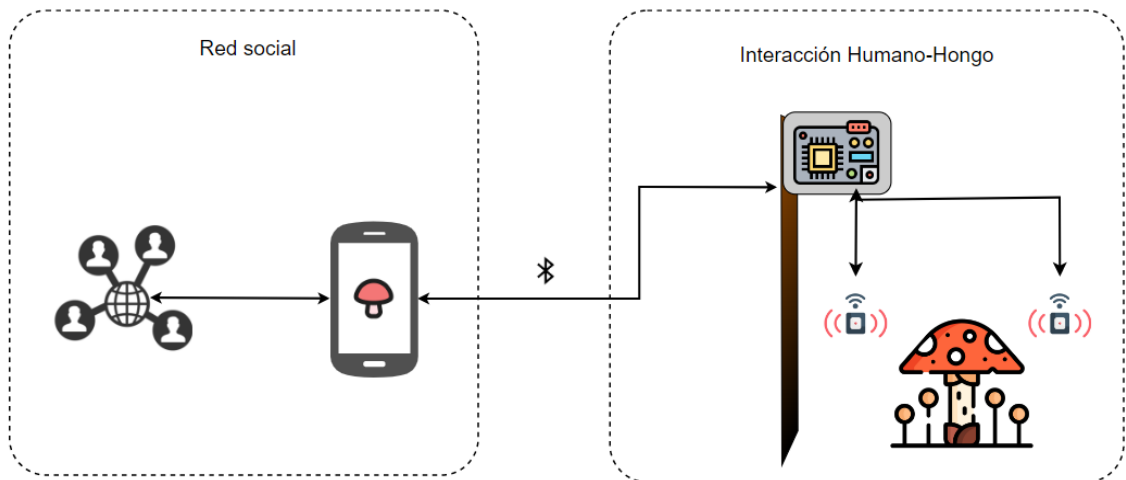


Figura 1. Esquema general del sistema.

Respectos a los impactos económico-sociales, se tiene un aumento de la concientización de la importancia del reino Fungi por parte de las personas. Esto implica que las personas que hayan formado parte de esta red tendrán una base mucho más informada que el resto acerca de qué especies de hongos existen en la zona, cuál es el rol biológico que cumplen en los ecosistemas, cómo protegerlos, etc. En cuanto a los impactos científico-tecnológicos está la validación de un modelo de conformación de red usado para la divulgación científica, que, si bien en este caso se enfoca el reino Fungi, se establecerían bases para replicarse en otras áreas de divulgación.

1.2 Objetivos

Objetivo general

Desarrollar una aplicación móvil que incorpore un modelo de conformación de red e integre lo físico y lo digital para fortalecer la relación Humano-Hongo y divulgar la importancia del reino Fungi a la comunidad.

Objetivos específicos

1. Desarrollar el modelo de conformación de red para el fortalecimiento de la relación entre humanos y hongos.
2. Integrar lo físico y lo digital para establecer un vínculo entre hongos y humanos estableciendo los dispositivos de cohabitación como puntos de inicio de esta red.
3. Establecer una experiencia de usuario interactiva y que permita la fácil divulgación del reino Fungi a través de la red.

2. MARCO TEÓRICO

2.1 Introducción

En este capítulo, se expone el marco teórico que respalda la investigación realizada para el desarrollo del proyecto. Con el objetivo de establecer una base sólida para este trabajo, se llevó a cabo una Revisión Sistemática de la literatura disponible, la cual aborda aspectos fundamentales relacionados con la interacción de la tecnología y el reino Fungi, así como el desarrollo de aplicaciones móviles para propósitos de divulgación. A continuación, se define un modelo de red, se presenta una contextualización del reino Fungi con el fin de explicar algunos conceptos aplicados en el proyecto. Por último, se aborda la integración de la aplicación móvil con sistemas embebidos.

2.2 Revisión Sistemática

La Revisión Sistemática es una técnica utilizada en el campo de la investigación científica para reunir y analizar toda la evidencia disponible sobre un tema específico. Esta técnica se utiliza para obtener una visión completa y precisa del estado actual del conocimiento sobre un tema en particular. El proceso de la Revisión Sistemática se lleva a cabo siguiendo un conjunto riguroso de pasos que incluyen la definición de una pregunta de investigación clara, la identificación de todos los estudios relevantes sobre el tema, la recopilación y análisis de los datos de esos estudios y la presentación de los resultados de forma clara y concisa.

2.2.1 Desarrollo de Revisión Sistemática

Objetivo y preguntas de revisión

El objetivo de la Revisión Sistemática en el contexto de este trabajo es producir una evaluación imparcial y objetiva de la evidencia disponible sobre el desarrollo de aplicaciones móviles para divulgar el reino Fungi o similares, de forma que esta información ayude a tomar decisiones informadas sobre el tema. En base a esto, se plantearon las siguientes preguntas de investigación:

1. ¿Qué tecnologías han sido utilizadas para el desarrollo de una aplicación móvil de divulgación del reino Fungi?
2. ¿Qué herramientas y tecnologías se han aplicado para la divulgación en espacios públicos como jardines botánicos?

2.2.2 Método de revisión

Metodología

Para llevar a cabo esta Revisión Sistemática, en primer lugar, se analizaron las preguntas de revisión para extraer las palabras clave y sus sinónimos del contexto de la investigación, y se las relacionó según corresponda para formular una cadena de búsqueda adecuada. A continuación, se seleccionaron las fuentes donde se realizaría la revisión y se llevó a cabo una selección de todos los artículos encontrados, con el objetivo de conservar los más relevantes para esta investigación, basándose en ciertos criterios arbitrarios que se detallarán más adelante. Finalmente, se extrajo la información siguiendo una serie de parámetros de interés. A continuación, se describe detalladamente el proceso de Revisión Sistemática.

Formulación cadena de búsqueda

Las palabras claves identificadas a partir de las preguntas de investigación son:

Mobile, "software development", smartphone, learn, gamification, teach, divulgation, outreach, dissemination, disclosure, fungi, "fungus kingdom", "fungus family", mushroom, "botanical garden".

Luego, se clasificaron estas palabras por similitud para estructurar la cadena de forma coherente, esta agrupación resultante queda como sigue.

- Mobile, software development, smartphone.
- Learn, gamification, teach, divulgation, outreach, dissemination, disclosure.
- Fungi, "fungus kingdom", "fungus family", mushroom, "botanical garden".

En base a esta agrupación de conceptos, el string de búsqueda resultante es:

(mobile OR "software development" OR smartphone)

AND

(learn OR gamification OR teach OR divulgation OR outreach OR dissemination OR disclosure)

AND

(fungi OR "fungus kingdom" OR "fungus family" OR mushroom OR "botanical garden")

Fuentes y estrategias de búsqueda

En la primera iteración del proceso se consideró buscar solamente en WoS (Web of Science), sin embargo, dado que se trata de un tema bastante específico, se obtuvieron muy pocos resultados, por lo que en la Revisión Sistemática final se optó por incluir las fuentes ACM (Association for Computing Machinery) y Scopus para explorar un rango

de búsqueda más amplio. Adicionalmente, se incluyó una fuente propia donde se incorporaron manualmente los artículos y Trabajos de Título proporcionados por los profesores patrocinante y co-patrocinante.

Además, luego de la primera iteración se tomó la decisión de incluir todas las secciones de los artículos, ya que inicialmente solo se consideraba el resumen y los resultados fueron poco relevantes. Como filtro general, se consideraron artículos desarrollados en los últimos cinco años.

Criterios de selección

Para seleccionar los artículos relevantes para este estudio, es decir, que incorporan el uso de tecnologías móviles aplicadas a la divulgación del reino Fungi y otros contextos relacionados, se definieron los siguientes criterios para la inclusión y exclusión (ver Tabla 1).

Tabla 1. Criterios de selección.

Criterio de inclusión	Criterio de exclusión
Aborda la divulgación del reino Fungi	No tiene relación con el contexto de divulgación
Implementa aplicaciones móviles para la divulgación	No aplica tecnologías
Implementa tecnologías web para la divulgación	

Selección final de artículos

Siguiendo los pasos previamente descritos, se obtuvieron un total de 72 artículos, de los cuáles se leyó el resumen y siguiendo los criterios de selección se descartaron 59, quedando solo 14 para la extracción de información. En la Figura 2, se muestra la cantidad de artículos seleccionados por cada una de las cuatro fuentes investigadas.

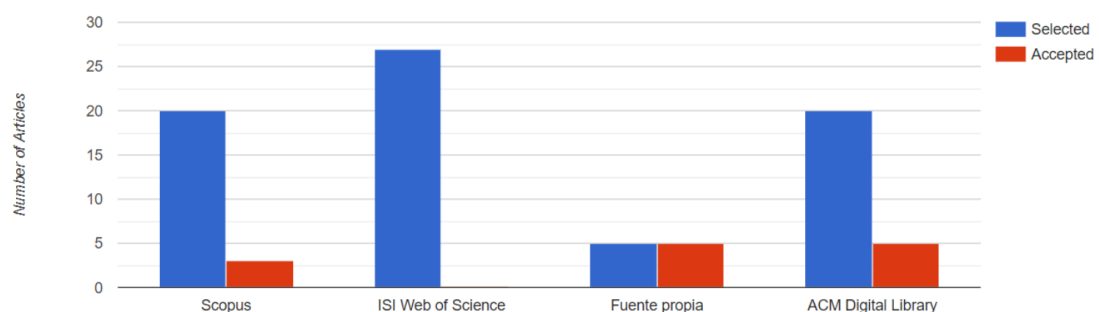


Figura 2. Selección de artículos por fuente.

En cuanto a la temporalidad, como se observa en la Figura 3, los artículos seleccionados son mayoritariamente del año 2022, registrando una tendencia creciente en cuanto a los dos años anteriores. Esto puede deberse a que la pandemia afectó esta clase de investigaciones y también a que este tipo de iniciativas están cobrando mayor interés en el último tiempo.

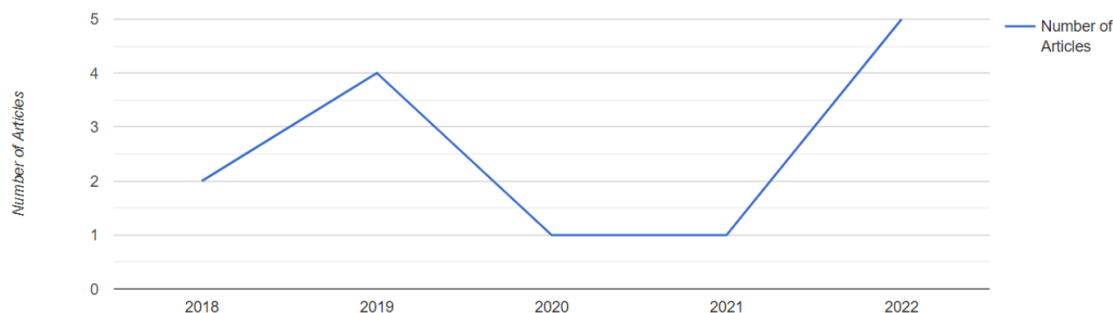


Figura 3. Selección de artículos por año.

Pauta de extracción

Para la extracción de la información más relevante, se definió una pauta de extracción en base a una serie de parámetros para realizar el proceso de forma estructurada y orientada al objetivo de esta investigación. La pauta de extracción es la siguiente:

- Modelo o solución propuesta.
- Herramientas y funcionalidades.
- Usos y experiencias.
- Tecnologías utilizadas.
- Hallazgos importantes.

2.2.3 Resultados

La información relevante de cada uno de los artículos seleccionados se estructura en la tabla que se muestra a continuación desde la Tabla 2 a la Tabla 13.

Tabla 2. Extracción de información del primer artículo.

Autor
Coelho, A., van Zeller, M., Cardoso, P., Santos, L., Vaz, R., & Raimundo, J. (2020).
Nombre del artículo
Gamifying the Museological Experience.
Modelo o solución propuesta
Se presentan tres diferentes propuestas de gamificación para mejorar la experiencia de los visitantes en lugares como museos o jardines botánicos. <ol style="list-style-type: none"> 1. Gamificación en museos: Se propone una aplicación móvil que incluye un juego basado en el concepto de colección de stickers para un álbum, incorporando realidad aumentada. 2. Juegos basados en localización en parques: Modelo de juego móviles con fines educativos y culturales que permiten a los usuarios explorar el mundo real mientras juegan. Se desarrolló un prototipo para un parque en Portugal donde se usa la posición del jugador en tiempo real usando GPS y mediante un mapa mostrar los puntos de interés. Uno de estos juegos tenía un modo de "búsqueda del tesoro". 3. Caso de un jardín botánico: Se trata de un videojuego móvil basado en localización donde el jugador va progresando a medida que encuentra puntos de interés (POI), siguiendo una secuencia específica entre ellos resolviendo retos. Estos retos pueden ser resueltos interactuando con el entorno.
Herramientas y funcionalidades
En general, dado que se exploran diferentes casos de estudio, los proyectos se basan en el uso de un smartphone con GPS y uso de otras tecnologías como realidad aumentada (AR).
Usos y experiencias
Todos estos proyectos están diseñados para mejorar la experiencia de los visitantes en espacios como museos, parques y jardines botánicos mediante el uso de tecnología.
Tecnologías utilizadas
No se hace mención específica de tecnologías de desarrollo, pero en general se hace uso del GPS de los smartphones, AR, microcontroladores, sensores, interfaces hápticas, etc.
Hallazgos importantes
Existe un gran potencial de la tecnología para mejorar las experiencias en el mundo real incorporando gamificación. Sin embargo, el uso de las tecnologías debe ser balanceado, de otra forma puede causar distracción.

Tabla 3. Extracción de información del segundo artículo.

Autor
Barrientos-Avendaño, E., Rico-Bautista, D., Coronel-Rojas, L. A., & Cuesta-Quintero, F. R. (2019).
Nombre del artículo
Jardín botánico: Prototipo de software para la gestión y divulgación de plantas nativas basado en código QR y realidad aumentada.
Modelo o solución propuesta
Sistema para monitorear el estado de las plantas de un jardín botánico y mejorar la experiencia de exploración y divulgación de las plantas nativas entre investigadores. Este sistema se piensa extender a futuro para visitantes casuales del jardín botánico y mejorar su experiencia mediante códigos QR y realidad aumentada.
Herramientas y funcionalidades
El sistema permite el registro de plantas, evaluar características detectadas por los investigadores y realizar consultas inteligentes con el objetivo de tomar decisiones en base a información de interés.
Usos y experiencias
Esta versión del sistema está diseñada para trabajadores del jardín e investigadores y les facilita la tarea de virtualización del jardín botánico y que a futuro se brinde una interfaz interactiva que mejore la experiencia de los visitantes del jardín botánico.
Tecnologías utilizadas
Para el desarrollo de este sistema se utilizó Android Studio, Blender, Unity y Vuforia. Por otro lado, se Implementó tecnología QR y realidad aumentada.
Hallazgos importantes
El uso eficiente de la tecnología permite agilizar en gran medida los procesos que se llevan a cabo en un jardín botánico y permite generar un sentido de pertenencia en los visitantes.

Tabla 4. Extracción de información del tercer artículo.

Autor
Nobnop, R., Wongwatkit, C., & Soponronnarit, K. (2018).
Nombre del artículo
A development of 3D augmented reality mobile application to facilitating ecotourism-based herbal learning in MFU botanical garden.
Modelo o solución propuesta
Una aplicación móvil para visitantes del jardín botánico MFU que incorpora realidad aumentada de gráficos y audio para mejorar el ecoturismo y la experiencia de la visita.
Herramientas y funcionalidades
La herramienta principal para poder utilizar esta aplicación es un smartphone que soporte AR. Con esta aplicación pueden escanear imágenes de plantas y obtener información interactiva al respecto creando un modelo 3D.
Usos y experiencias
Esta aplicación está destinada al uso cotidiano de los visitantes del jardín botánico MFU brindando una Experiencia mucho más enriquecedora.
Tecnologías utilizadas
Para el desarrollo de la aplicación se usó Unity con C# y para el modelado 3D se utilizó Autodesk Maya. También se aplicó tecnología de realidad aumentada.
Hallazgos importantes
Presentar la información de la manera interactiva que provee la realidad aumentada puede favorecer mucho el proceso de aprendizaje logrando que las personas se diviertan mientras aprenden.

Tabla 5. Extracción de información del cuarto artículo.

Autor
Winter, A., Pedro, E., Ślasko, J., Battaglini, J., Faelker, M., Kivipelto, R., ... & Guedes, P. (2019).
Nombre del artículo
Waste to Fungi: An EPS@ ISEP 2019 Project
Modelo o solución propuesta
Sistema de cultivo y monitoreo de hongos mediante el uso de residuos de granos de café siguiendo un modelo de negocios de economía circular sustentable.
Herramientas y funcionalidades
Para los sistemas de cultivos se usan recipientes plásticos reutilizados, y dentro de estos, una cámara de monitoreo con todo el software y hardware para medir la temperatura, humedad y crecimiento.
Usos y experiencias
La realización de este proyecto permitió crear un nuevo modelo de negocio basado en economías circulares que puede ser aplicado por cualquier aficionado al cultivo de hongos y que brinda a los cultivadores una experiencia mucho más sustentable y amigable con el medio ambiente.
Tecnologías utilizadas
Para el monitoreo se usa un microcontrolador conectado a diferentes sensores e incluyendo un módulo wifi para enviar datos a la nube. Por lado de la interfaz de usuario, estos datos se pueden monitorear a través de una aplicación móvil.
Hallazgos importantes
Con este trabajo se pudo validar un modelo de cultivo de ciertas especies de hongos donde se implementa una economía circular sustentable que permita reducir los residuos del proceso al mínimo.

Tabla 6. Extracción de información del quinto artículo.

Autor
Liu, J., Byrne, D., & Devendorf, L. (2018)
Nombre del artículo
Design for collaborative survival: An inquiry into human-fungi relationships.
Modelo o solución propuesta
En este trabajo se busca co-diseñar interfaces de usuario innovadoras donde se introduce el concepto de supervivencia colaborativa y se presentan tres prototipos tecnológicos distintos a base de sensores para forjar relaciones mucho más directas entre humanos y hongos.
Herramientas y funcionalidades
El primer dispositivo (HSI) consiste en un guante equipado con sensores en los dedos donde a través del tacto y la inserción de los dedos en el terreno se pueda detectar información acerca del sustrato (suelo, materia orgánica, etc.) que apoya el crecimiento de los hongos. El diseño se basa en un sensor de humedad. El segundo prototipo se denomina Data HarVest es una prenda usable que permite registrar vía GPS avistamientos de hongos en crecimiento para ir generando un mapa. El último dispositivo consiste en un bastón de trekking modificado para recolectar esporas durante un trayecto para posteriormente estudiar las especies de hongos que coexisten en ese ecosistema.
Usos y experiencias
Su uso está principalmente pensado para aficionados de la micología, aunque puede extenderse a cualquier persona. La finalidad de estos dispositivos es establecer nuevas formas de relacionarse con el entorno y particularmente con los hongos.
Tecnologías utilizadas
No se habla en detalle, pero se infiere que para la construcción de estos dispositivos se usan microcontroladores conectados a diversos módulos y sensores.
Hallazgos importantes
Se da a conocer el gran potencial que hay en el uso de la tecnología para aportar al desarrollo de interacciones entre humanos y hongos y al progreso de HCI (Human-Computer Interface).

Tabla 7. Extracción de información del sexto artículo.

Autor
Mauricio Yagui, M. M., Monsores Passos Maia, L. F., Oliveira, J., & Vivacqua, A. (2019).
Nombre del artículo
A Crowdsourcing Platform for Curating Cultural and Empirical Knowledge. A Study Applied to Botanical Collections.
Modelo o solución propuesta
Una plataforma que implementa un conjunto de procesos, actividades y prácticas para el apoyo colaborativo de la divulgación entre expertos y visitantes en el ámbito de la botánica. Para ello se propone un esquema de bases de datos RDFS de conocimiento cultural que evolucione en el tiempo.
Herramientas y funcionalidades
La herramienta permite a los visitantes enviar contenido y registrar conocimiento empírico de un tema. Además, permite a especialistas (investigadores o usuarios con conocimientos en un dominio específico) recomendar información relacionada.
Usos y experiencias
Este sistema tiene aplicaciones en cualquier ámbito de divulgación donde existan espacios físicos donde convivan visitantes ocasionales y especialistas.
Tecnologías utilizadas
El sistema fue escrito en PHP y la librería open source EasyRDF para backend y Bootstrap para frontend. Para las bases de datos se usó Apache Jena Fuseki. También se utiliza tecnología QR.
Hallazgos importantes
Un modelo de divulgación con una arquitectura capaz de evolucionar en el tiempo aplicable a diversos ámbitos.

Tabla 8. Extracción de información del séptimo artículo.

Autor
Islam, M. A., Islam, M. A., Miah, M. S. U., & Bhowmik, A. (2022)
Nombre del artículo
An automated monitoring and environmental control system for laboratory-scale cultivation of oyster mushrooms using the Internet of Agricultural Thing (IoAT).
Modelo o solución propuesta
Sistema de control automatizado de hongos ostra utilizando IoAT para fines de cultivo a pequeña escala de bajo costo, aunque con potencial escalable.
Herramientas y funcionalidades
El sistema se acopla a un ambiente de cultivo controlado y mide el nivel de humedad, temperatura y luz en tiempo real para que las especies se mantengan en buen estado. Los datos pueden ser monitoreados con un smartphone.
Usos y experiencias
El sistema está pensado para el cultivo de hongos ostra u otras especies de manera indoor donde el granjero puede monitorear los parámetros en tiempo real. Se dice que la propuesta tiene potencial para un nivel de producción mayor.
Tecnologías utilizadas
Su utiliza un microcontrolador ESP-8266 con módulo wifi conectado con sensores DHT11 de humedad y temperatura. En el lado web se utiliza un servidor XAMPP y Apache.
Hallazgos importantes
Se logró construir un sistema de control y monitoreo del cultivo de hongos exitoso y a muy bajo costo, lo que reduce el gran potencial de esta propuesta para seguir escalando en la cantidad de parámetros medidos y la capacidad del sistema.

Tabla 9. Extracción de información del noveno artículo.

Autor
Chirica, D. B., & Iftene, A. (2021).
Nombre del artículo
Enhancing the Visit of the Botanical Garden of Iasi with the Help of Smart Devices and Augmented Reality. (2021)
Modelo o solución propuesta
Una aplicación móvil con cuatro diferentes componentes para mejorar la experiencia de los visitantes del jardín botánico incorporando realidad aumentada.
Herramientas y funcionalidades
El primer componente tiene la funcionalidad de contar los pasos del usuario usando el sensor incorporado en los smartphones. Esto se almacena en la base de datos de usuario para mostrar la cantidad de pasos por día. El segundo es un juego tipo quiz para identificar especies de plantas obteniendo puntaje. El otro es un juego tipo "búsqueda del tesoro" donde se deben encontrar diversas plantas en el jardín botánico pudiendo recibir algunas pistas. También se puede navegar al perfil de usuario donde toda la información más relevante se despliega.
Usos y experiencias
Esta aplicación tiene por principal objetivo mejorar la experiencia de los visitantes del jardín botánico mediante el uso de la tecnología. Gracias a esto, los usuarios pueden aprender de forma mucho más dinámica.
Tecnologías utilizadas
Esta aplicación tiene por principal objetivo mejorar la experiencia de los visitantes del jardín botánico mediante el uso de la tecnología. Gracias a esto, los usuarios pueden aprender de forma mucho más dinámica.
Hallazgos importantes
Esta propuesta logra combatir el sedentarismo que existe en la juventud con muy buenos resultados donde pueden aprender mientras realizan actividad física.

Tabla 10. Extracción de información del décimo artículo.

Autor
Guillen Lozano, F. A., & Zapata Galarza, K. S. (2022).
Nombre del artículo
Aplicación móvil para el aprendizaje del reino fungi en bosques de selva alta con realidad aumentada, gamificación y microlearning.
Modelo o solución propuesta
Una aplicación para la divulgación del reino Fungi con el objetivo de medir el impacto en los usuarios mediante encuestas y análisis estadístico lo cual es el foco de la tesis.
Herramientas y funcionalidades
La aplicación consta de diversos módulos para enseñar acerca del reino fungi como por ejemplo una función de identificación de hongos, realidad aumentada, algunos juegos incorporados y obtención de Logros.
Usos y experiencias
Esta aplicación está pensada para consumidores de hongos sirviendo como herramienta para educarlos acerca de las especies de hongos comestibles y venenosas.
Tecnologías utilizadas
Para el desarrollo de la aplicación se usó Android Studio con Java como lenguaje, Blender, Autodesk y Firebase para el backend.
Hallazgos importantes
Se obtuvo evidencia de que el uso de esta aplicación incrementó el conocimiento del reino Fungi en los usuarios, así como la motivación de aprender, obteniendo un grado satisfacción bastante alto.

Tabla 11. Extracción de información del onceavo artículo.

Autor
Ibrahim, N. Z., Osman, R., Ali, A. M., Hamid, N. H. A., Nordin, S., Shahimi, S., ... & Razak, F. A. (2022).
Nombre del artículo
Integrating Augmented Reality in Learning Plants.
Modelo o solución propuesta
Una aplicación móvil que usa realidad aumentada para enseñar acerca de las plantas.
Herramientas y funcionalidades
Mediante una función de escanear, el usuario puede leer una imagen y visualizar videos y modelos 3D con realidad aumentada, también incluye la función para explorar las diferentes especies de plantas y aprender de ellas.
Usos y experiencias
Esta aplicación permite enseñar a cualquier persona interesada de una forma mucho más dinámica acerca de las plantas.
Tecnologías utilizadas
Para el desarrollo se utilizó Android Studio, Unity 3D, Firebase para la base de datos y usando Blender y Vuforia para la implementación de los modelos de realidad aumentada.
Hallazgos importantes
Se aplicaron encuestas a usuarios donde se evidenció que el uso de realidad aumentada es bien recibido y brinda buenos resultados.

Tabla 12. Extracción de información del doceavo artículo.

Autor
Postolache, S., Torres, R., Afonso, A. P., Carmo, M. B., Cláudio, A. P., Domingos, D., ... & Redweik, P. (2022).
Nombre del artículo
Contributions to the design of mobile applications for visitors of Botanical Gardens.
Modelo o solución propuesta
No aplica.
Herramientas y funcionalidades
Se hace un análisis de requisitos funcionales y no funcionales para aplicaciones móviles destinadas a la exploración y divulgación de jardines botánicos.
Usos y experiencias
Este estudio sirve como guía para desarrolladores de software que se embarcan en desarrollar aplicaciones móviles para jardines botánicos enmarcando la vanguardia tecnológica y buenas prácticas de calidad.
Tecnologías utilizadas
En el artículo se mencionan algunas de las tecnologías más atractivas para implementar como son códigos QR, GPS, NFC, realidad aumentada, etc.
Hallazgos importantes
Se entregan diversas contribuciones para desarrollar aplicaciones móviles en jardines botánicos aplicables a otros contextos como museos o parques.

Tabla 13. Extracción de información del treceavo artículo.

Código del artículo
Ulloa, R. (2019).
Nombre del artículo
Desarrollo de una plataforma web y aplicación móvil para el control automatizado de un invernadero.
Modelo o solución propuesta
Un sistema de monitoreo de invernadero conformado por una plataforma web, aplicación móvil y un microcontrolador con diferentes sensores que permiten controlar el invernadero de forma automatizada.
Herramientas y funcionalidades
El sistema permite monitorear mediante la aplicación móvil las principales variables para el control de un invernadero como son la humedad ambiental, temperatura y humedad del suelo.
Usos y experiencias
Este sistema está pensado para granjeros y personas a cargo de invernaderos para facilitar su trabajo gracias a la tecnología con un enfoque sustentable.
Tecnologías utilizadas
Para el desarrollo de este sistema se utilizó Basic4Android, servidor web Apache, SQLite y MySQL para la base de datos, PhpMyAdmin, XAMPP, microcontrolador IOIO-OTG y diversos sensores para medir humedad y temperatura.
Hallazgos importantes
La implantación de este sistema tuvo una recepción positiva por parte del usuario, el cual logró usarla efectivamente lo que permitió facilitar su trabajo y usar los recursos más eficientemente.

2.1.4 Principales hallazgos

Luego de extraer la información más relevante para la investigación se pueden establecer los principales hallazgos con el fin de responder las preguntas inicialmente propuestas. En general, se evidenció que el trabajo relacionado a la divulgación del reino Fungi es bastante acotado, sin embargo, en otros contextos como es el caso de plantas y árboles existe un estado del arte mucho más desarrollado donde muchos de los modelos propuestos pueden extrapolarse al contexto Fungi dadas las similitudes. Algunos de los hallazgos más importantes son descritos a continuación.

Sistemas embebidos

En diversos artículos se expone el uso de microcontroladores para vincular lo físico con lo digital, lo cual es uno de los objetivos de este Trabajo de Título. El uso de microcontroladores ofrece una amplia gama de funcionalidades para un proyecto de este tipo, en la literatura analizada se exponen, por ejemplo, el monitoreo de hongos o plantas mediante diferentes tipos de sensores. También, en algunos artículos se estudian los parámetros que son interesantes para medir en hongos, ya que estos organismos suelen

necesitar condiciones bastante específicas para vivir. Estos parámetros son la humedad ambiental, humedad del suelo, temperatura y nivel de pH. Algunos de los microcontroladores usados son placas Arduino, IOIO y Raspberry.

Gamificación

Otro aspecto muy común entre trabajos explorados es el desarrollo de aplicaciones móviles con un enfoque de gamificación para atraer la atención de las personas mediante juegos integrados o diferentes funcionalidades que emulen esto. La gamificación es el uso de elementos y técnicas propias de los juegos en contextos que no son estrictamente juegos, con el fin de aumentar la participación, la motivación y el compromiso de las personas. De esta forma, se logra un aprendizaje más entretenido que en general tiene buena recepción por parte de los usuarios.

Códigos QR

Una tecnología bastante conocida hoy en día son los códigos QR y pueden integrarse de manera muy sencilla en aplicaciones móviles destinadas a la enseñanza y divulgación. Las ventajas que otorga implementar esta tecnología es que los usuarios pueden elegir acerca de lo que quieren saber más sin sobrecargarse de información.

Realidad aumentada

Otra tecnología que se implementó en varios de los trabajos revisados es la realidad aumentada (AR). Consiste en brindar una interfaz que permita conectar el mundo real con el digital, esta tecnología se ha vuelto una tendencia en los últimos años porque los smartphones son ideales para este tipo de interacción siendo accesible para todo el mundo. Cuando se usa en el contexto de divulgación brinda una experiencia mucho más inmersiva y enriquecedora a las personas, algo que se alinea muy bien con el objetivo de este trabajo.

2.1.5 Conclusiones de la Revisión Sistemática

En conclusión, se puede decir que la Revisión Sistemática permitió dilucidar que, aunque el trabajo relacionado a la divulgación del reino Fungi es limitado, es un campo que ha ido cobrando mayor relevancia en los últimos años. Además, existen diversas tecnologías y enfoques que pueden ser aplicados en este contexto. Entre los principales hallazgos se destacan el uso de sistemas embebidos como microcontroladores, la gamificación a través de aplicaciones móviles, la implementación de códigos QR y la realidad aumentada. Estas herramientas pueden resultar muy útiles para atraer la atención del público y ofrecer una experiencia de aprendizaje más enriquecedora e inmersiva. En definitiva, la aplicación de estas tecnologías puede contribuir significativamente a la divulgación y concientización sobre la importancia de los hongos y su papel fundamental en los ecosistemas

2.3 Modelo de conformación de red

Las redes sociales forman parte del día a día de la mayoría de las personas en la actualidad llegando a forjarse una cierta dependencia hacia ellas. Si bien existen algunos aspectos negativos asociados al uso de estas plataformas como su uso excesivo o la desinformación, también existen beneficios que pueden ofrecer las redes sociales y que son particularmente aplicables al contexto de la divulgación científica. Uno de ellos es que las redes promueven la interacción entre las personas facilitando el intercambio de información de manera ágil. Las redes sociales pueden aportar al desarrollo de la comunidad de especialistas y aficionados al estudio de los hongos a nivel local y potencialmente regional, fomentando la colaboración y difusión de investigaciones e hitos importantes. Asimismo, permitir que personas no tan involucradas con el reino Fungi puedan formar parte de esta comunidad. Otro beneficio es que las redes sociales y las aplicaciones móviles en general aprovechan la vinculación que tienen las personas hoy en día con los dispositivos móviles, especialmente la población juvenil.

El modelo de red que se implementará se conoce como red social vertical, este tipo de redes se basan en una temática concreta y forman comunidades a partir de la interacción entre usuarios con intereses afines. Las redes sociales verticales se caracterizan por ofrecer a los usuarios una experiencia mucho más personalizada y productiva en comparación a su contraparte, las redes horizontales, las cuales tienden a ser mucho más genéricas (Armetrics, 2020). En el caso particular de este proyecto, la red social vertical se especializa en el reino Fungi y se busca brindar una experiencia especializada e interactiva a los usuarios.

2.4 Contextualización del reino Fungi

Los hongos son organismos mucho más complejos de lo que comúnmente se piensa. A pesar de que son comparados con las plantas, están taxonómicamente más emparentados con los animales. Existen diferentes tipos de hongos, aunque los más comunes son los denominados filamentosos y son aquellos que producen las setas en su fase reproductiva. Es común creer que las setas que crecen en el suelo son el hongo en sí, sin embargo, estas solo representan el cuerpo fructífero o reproductivo del hongo, ya que bajo la tierra se encuentra el micelio que es una compleja estructura en forma de red (Rothschuh, 2021), como se observa en la Figura 4.

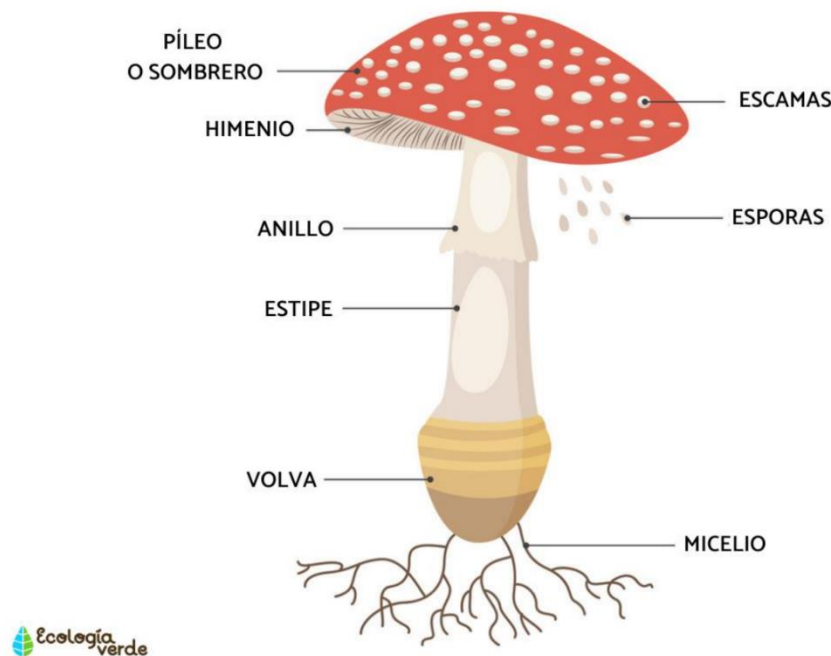


Figura 4. Estructura de un hongo.¹

Los hongos son capaces de subsistir en diferentes ecosistemas, sin embargo, requieren que se den una serie de condiciones idóneas en el ambiente para que puedan proliferar y desarrollarse normalmente. Según lo consultado en una reunión con la especialista en hongos y profesora Auxiliar de la Facultad de Ciencias Agrarias y Alimentarias de la Universidad Austral de Chile, existen dos variables de particular interés para medir mediante sensores en un hongo y estos son el porcentaje de humedad atmosférica y la temperatura en torno al cuerpo fructífero del hongo. Adicionalmente, la doctora mencionó que el pH también puede ser útil medirse en caso de ser posible. Estos parámetros pueden variar según la especie y el poder medirlos puede ayudar al estudio de estas. A continuación, se describen estos parámetros con mayor detalle.

Humedad

La humedad ambiental consiste en la cantidad de vapor de agua presente en el aire y es uno de los factores más importantes para el crecimiento y reproducción de los hongos. En general, se estima que la humedad en el ambiente debe tener valores bastante elevados (aproximadamente 90%) para el desarrollo de los hongos y si se detecta un nivel inferior

¹ Roths Schuh, U. 2021. Estructura de los hongos. [Figura]. Recuperado de: <https://www.ecologiaverde.com/estructura-de-los-hongos-3676.html>

al 13%, se denomina una condición de bajas probabilidades de crecimiento y proliferación fúngica (Suconel, 2013).

Temperatura

La temperatura representa otro factor determinante en el desarrollo de los hongos y se estima que el rango óptimo de esta variable se encuentra entre los 25 y 30°C, pudiendo llegar hasta los 40 y 45°C. sin embargo, se tiene conocimiento de excepciones en especies que pueden subsistir entre los 0 y 55°C sin problemas (Suconel, 2013). Algunas de las especies endémicas del país y particularmente de la región forman parte de este último grupo, ya que se desarrollan y fructifican en las temporadas de otoño e invierno, las cuales se caracterizan por las bajas temperaturas.

pH

Es importante también mencionar algunos factores químicos como el pH (potencial de hidrógeno). Esta medida hace referencia al grado de alcalinidad o acidez de una disolución. Los hongos pueden tolerar un intervalo bastante amplio de pH (2.5 – 7.5). De manera general, soportan mejor el medio ácido que el alcalino. Sin embargo, es importante mencionar que los mismos hongos son capaces de alterar el pH utilizando los desechos del ecosistema (Gimeno, 2002).

2.5 Integración con sistemas embebidos

Como se mencionó previamente, el desarrollo de los hongos está influenciado por diversos factores determinantes. Sin embargo, es importante destacar que los rangos normales de estos parámetros pueden variar según la especie en estudio. Por esta razón, como parte integral de este sistema, se propone la integración de un dispositivo que consta de un microcontrolador conectado a varios sensores de humedad, temperatura y posiblemente pH. Este dispositivo permitirá a los usuarios realizar mediciones y compartir información con la comunidad, brindando así una experiencia más interactiva y fomentando el avance en el estudio de las especies endémicas.

3. METODOLOGÍA

3.1 Introducción

Para el desarrollo de este proyecto se optó por las metodologías ágiles principalmente por la naturaleza de innovación que se propone y además, por el enfoque flexible que promueven estas metodologías ante un escenario incierto con requisitos cambiantes, lo cual se adaptaba muy bien a este proyecto en particular, ya que inicialmente habían diversos factores relevantes como son el alcance y la escala sujetos a una postulación de fondos Fondart.

3.2 Toma de requisitos

La toma de requisitos efectiva es un elemento fundamental para el éxito de un proyecto de desarrollo de software. Proporciona una base sólida para la creación de una aplicación móvil que se ajuste a las expectativas de los usuarios y genere una experiencia satisfactoria. A través de una comprensión clara de los requisitos, es posible diseñar y desarrollar una aplicación que aborde los problemas y necesidades reales de los usuarios, ofreciendo una solución que sea útil, usable y atractiva.

La metodología utilizada para la toma de requisitos se basó en entrevistas a stakeholders clave y en el análisis de los resultados obtenidos. El objetivo principal fue recopilar la información necesaria para diseñar y desarrollar una aplicación móvil que cumpla con las expectativas de los usuarios y genere una experiencia satisfactoria.

El proceso se dividió en las siguientes etapas:

- Definición del objetivo: Se estableció el objetivo de la toma de requisitos, que consistió en obtener información para la creación de una aplicación móvil enfocada en la interacción entre humanos y hongos, abordando los problemas y necesidades reales de los usuarios.
- Identificación de stakeholders: Se identificaron dos stakeholders clave para las entrevistas: el director del proyecto y experto en diseño, y la micóloga, experta en hongos. Estos stakeholders representaron perspectivas diferentes pero complementarias para comprender los requisitos del proyecto.
- Entrevistas a los stakeholders: Se realizaron entrevistas individuales a cada uno de los stakeholders identificados. Durante las entrevistas, se obtuvo información sobre los antecedentes del proyecto, las expectativas, sugerencias y requisitos específicos desde sus respectivas perspectivas.
- Análisis de resultados: Se llevó a cabo un análisis de los resultados de las entrevistas para identificar requisitos comunes y específicos mencionados por los

stakeholders. Se buscaron patrones y puntos en común para obtener una visión clara de los requisitos que deben ser considerados en el diseño y desarrollo de la aplicación móvil.

- Refinamiento de requisitos: Con base en el análisis de los resultados, se realizaron ajustes y refinamientos en los requisitos identificados. Se tuvieron en cuenta las necesidades de diseño atractivo, usabilidad y las restricciones relacionadas con la complejidad de los hongos y su fructificación estacional.

Modelo de entrevistas

Para el desarrollo de las entrevistas se elaboró una lista de preguntas empeñadas en conocer un poco acerca de los entrevistados en cuanto a su ocupación y rol que desempeñan en el proyecto. Luego, se definen las preguntas enfocadas en conocer la perspectiva de cada uno para esta aplicación móvil, así como sus expectativas. En el caso de la experta en hongos, se incluyeron preguntas más especializadas. A continuación, se listan las principales preguntas:

- ¿Cuál es tu ocupación o profesión actualmente?
- ¿Cuál es el rol que cumple en el proyecto Hongos y Humanos?
- ¿Cuál es el perfil de las personas que se espera que interactúen con estos dispositivos?
- ¿Piensas que hay un potencial tecnológico al que se le puede sacar provecho con esta aplicación móvil para enriquecer este proyecto?
- ¿Cuáles son tus expectativas de esta aplicación móvil?
- ¿Tienes ideas en cuanto a formas de interacción entre hongos y humanos que pueda facilitar esta aplicación?

3.3 Diseño

Design Thinking

Design Thinking o pensamiento de diseño consiste en una metodología que facilita el diseño de productos o servicios de todo tipo teniendo como motor la innovación y creatividad para los resultados. Es un proceso iterativo y consta de diferentes etapas (ver Figura 5), comenzando por la empatía donde se busca entender las necesidades, deseos y limitaciones de los usuarios, esto es muy importante en Design Thinking, ya que el diseño siempre es centrado en el usuario. La siguiente etapa consiste en la definición del

problema. A continuación, sigue la etapa de ideación, donde surgen diversas soluciones creativas. Luego de seleccionar las mejores ideas, se procede a construir prototipos. Por último, se concluye con la prueba y validación de estas (ITMadrid, 2020).



Figura 5. Fases del Design Thinking.²

Esta metodología se adapta muy bien a este proyecto donde las componentes de diseño, innovación y creatividad tienen gran importancia para lograr los objetivos propuestos. Además, es un proyecto de origen interdisciplinario donde existe colaboración de stakeholders pertenecientes a diferentes áreas de conocimiento como informática, diseño y micología.

Durante la primera fase del proyecto, se aplicó esta metodología con el objetivo de comenzar a definir el diseño de la aplicación. Esto incluyó la propuesta de un modelo de red, así como la identificación de funcionalidades y comportamientos básicos. Además, se buscó conocer las principales dificultades y limitaciones que pudieran surgir. Para lograr esto, se llevaron a cabo reuniones periódicas con los profesores patrocinante y copatrocinante, y se tomaron en cuenta los resultados de las entrevistas realizadas a stakeholders relevantes.

3.4 Implementación

Scrum

Para la implementación del sistema se aplicó la metodología Scrum, la cual consiste en un marco de trabajo ágil utilizado para la gestión y desarrollo de proyectos. Se basa en la idea de que un equipo autoorganizado y multidisciplinario puede trabajar de manera efectiva y eficiente para lograr los objetivos propuestos.

El proceso de Scrum se divide en ciclos cortos e iterativos llamados Sprints, que generalmente duran de una a cuatro semanas. Durante cada sprint, el equipo de trabajo se enfoca en completar una serie de tareas planificadas en un artefacto denominado Sprint

² ITMadrid. (2020). Qué es y para qué sirve el Design Thinking. [Figura]. Recuperado de: <https://www.itmadrid.com/que-es-y-para-que-sirve-design-thinking/>

Backlog. El progreso se mide en reuniones regulares, donde los miembros del equipo informan sobre los avances realizados y las tareas pendientes.

El objetivo de Scrum es permitir una mayor flexibilidad y adaptabilidad a los cambios en los requerimientos del proyecto, así como mejorar la calidad del producto final. Scrum también se enfoca en fomentar la colaboración entre los miembros del equipo, el cliente y otros stakeholders del proyecto, para asegurar la eficacia en el cumplimiento de los objetivos del proyecto (Schwaber & Sutherland, 2020).

Scrum fomenta la transparencia y la colaboración a través de la comunicación regular y abierta entre los miembros del equipo, el cliente y otros stakeholders relevantes. Esto asegura que todos tengan una comprensión clara de los objetivos del proyecto y puedan trabajar en conjunto para lograrlos. También se alienta la retroalimentación constante, lo que permite realizar ajustes a medida que se obtiene más información y se entienden mejor las necesidades del cliente.

4. ARQUITECTURA DE SOFTWARE

4.1 Introducción

En el marco del desarrollo de software, la arquitectura desempeña un papel fundamental en la creación de aplicaciones robustas, escalables y de fácil mantenimiento. La elección y diseño de una arquitectura adecuada resulta crucial para garantizar la eficiencia y calidad del software desarrollado. En este capítulo, se abordará detalladamente la arquitectura de software utilizada en el desarrollo de la aplicación, la cual sigue un modelo cliente-servidor y se basa en un enfoque de "Arquitectura Limpia" o "Clean Architecture" para asegurar la calidad y modularidad del código, así como facilitar el mantenimiento y la evolución de la aplicación. Esta arquitectura se basa en los principios de DDD (Domain Driven Design) favoreciendo la separación de responsabilidades, independencia de frameworks y dependencias externas. Asimismo, en este capítulo se mencionan todas las tecnologías involucradas en el desarrollo del sistema.

4.2 Tecnologías y herramientas utilizadas

4.2.1 Android

Android es un sistema operativo basado en Linux desarrollado por Google en 2003. Se caracteriza por ser de código abierto, lo que significa que su código fuente está disponible públicamente y puede ser utilizado y modificado por cualquier desarrollador interesado en hacerlo. Esta característica ha permitido que Android se convierta en uno de los sistemas operativos móviles más populares del mundo (Android Authority, 2022). De hecho, se estima que Android posee un 68,79% de la cuota de mercado de sistemas operativos móviles en el periodo abril 2022 – abril 2023 (GlobalStats' Statcounter, 2023).

4.2.2 Expo

Es una plataforma que permite desarrollar aplicaciones móviles multiplataforma utilizando JavaScript y React Native. React Native es un marco de desarrollo de aplicaciones móviles que utiliza JavaScript como lenguaje de programación y permite crear aplicaciones para iOS y Android de forma nativa. Si bien estas tecnologías permiten el desarrollo de aplicaciones nativas multiplataforma, se desarrolló solo para sistemas Android ya que en el caso de sistemas IOS se requiere incurrir en gastos adicionales para poder desplegar la aplicación móvil, además, para objeto de este proyecto, solo se desarrolla un prototipo.

Expo simplifica el proceso de desarrollo de aplicaciones móviles al proporcionar una serie de características y servicios preconstruidos que son comunes en muchas aplicaciones, como el acceso a la cámara, el uso de notificaciones push y la geolocalización. Además,

Expo cuenta con una serie de herramientas y bibliotecas adicionales que facilitan el desarrollo, la depuración y el despliegue de aplicaciones móviles.

Una de las principales ventajas de Expo es que no requiere una configuración compleja para comenzar a desarrollar una aplicación. Expo también proporciona un cliente móvil llamado Expo Go, que permite probar y compartir rápidamente las aplicaciones que se están desarrollando sin necesidad de configurar entornos de desarrollo separados (Expo, 2020).

4.2.3 Express JS

Es un framework de desarrollo web basado en Node JS ampliamente utilizado hoy en día. Proporciona una capa de abstracción y funcionalidades adicionales para facilitar la creación de aplicaciones web y APIs (Interfaces de programación de aplicaciones) de manera más rápida y sencilla.

Express se destaca por su enfoque minimalista y su naturaleza no opinante, lo que significa que proporciona una estructura básica y flexible, permitiendo a los desarrolladores tener un mayor control sobre la arquitectura y las decisiones de diseño de sus aplicaciones.

Algunas de las características clave de Express.js incluyen enrutamiento sencillo, manejo de solicitudes y respuestas HTTP, middleware para agregar funcionalidades adicionales, y una comunidad activa que proporciona una amplia gama de complementos y módulos para extender la funcionalidad básica del marco (MDN web docs, 2023).

4.2.4 MariaDB

MariaDB es uno de los sistemas de base de datos relacionales más populares en la actualidad. Fue creado por los desarrolladores originales de MySQL y es de código abierto. Comparte muchas similitudes con MySQL en términos de estructura de base de datos, comandos y funcionalidades. Esto significa que las aplicaciones y los sistemas diseñados para trabajar con MySQL pueden migrar fácilmente a MariaDB sin muchos cambios en el código.

La ventaja de MariaDB radica en su enfoque de ser una base de datos de código abierto y comunitaria, donde se fomenta la colaboración y se permite la participación de la comunidad en su desarrollo y mejora. También ha introducido nuevas características y mejoras sobre MySQL, como rendimiento mejorado, capacidad de almacenamiento en caché, optimización de consultas y mayor estabilidad (MariaDB Foundation, s.f.).

4.2.5 Docker

Consiste en una plataforma de código abierto que permite la creación, implementación y ejecución de aplicaciones mediante contenedores estandarizados. Un contenedor es una unidad liviana y portátil que encapsula todo lo necesario para que una aplicación se ejecute de manera independiente, incluyendo el código, las bibliotecas, las dependencias y las configuraciones del sistema. De esta forma se simplifica todo el proceso de despliegue de los servicios al crear ambientes de desarrollo consistentes y portátiles (IBM, 2013).

4.2.6 Ngrok

Ngrok es una herramienta de código abierto que facilita la creación de túneles seguros a través de redes públicas. Su objetivo principal es permitir la exposición segura de servidores locales o aplicaciones que se ejecutan en máquinas locales hacia Internet. En el caso de este proyecto, se utilizó Ngrok para implementar y desplegar la API REST de manera segura.

4.2.7 Arduino

Consiste en una plataforma de código abierto basada en software y hardware fácil de usar que permite el desarrollo de proyectos electrónicos interactivos.

El hardware principal de Arduino es una placa con un microcontrolador, que es el cerebro del sistema. Esta placa está diseñada para ser fácilmente programable y se puede conectar a diversos componentes electrónicos, como sensores, pantallas, motores y otros dispositivos.

El software de Arduino proporciona un entorno de desarrollo integrado (IDE) que facilita la escritura y carga del código en la placa Arduino. El lenguaje de programación utilizado se basa en C/C++, pero se proporcionan librerías y funciones simplificadas que permiten a los usuarios programar de manera más sencilla sin tener un conocimiento profundo de programación (Arduino Documentation, 2023).

En el contexto de este proyecto, se utilizó para el desarrollo del dispositivo de interacción entre humanos y hongos, el cual permite medir la temperatura y humedad de un hongo poder registrarlos en la aplicación móvil. Para ello se utiliza una placa Arduino UNO, un módulo Bluetooth HC-06 para la conexión y sensores de temperatura y humedad.

4.3 Caracterización de la arquitectura

En cuanto al diseño de la arquitectura, todo el sistema se basa en un modelo cliente-servidor, la lógica del sistema se separa en tres módulos principales, la capa de

presentación o frontend (cliente), el lado del servidor o backend, y por último, la capa denominada “Interacción Humano-Hongo” que se conforma por un microcontrolador conectado a sensores que permite capturar métricas de un hongo. A continuación, se describen estos tres módulos principales.

4.3.1 Frontend

El frontend del sistema engloba todo lo relacionado con la presentación de la aplicación móvil y tiene la función de proporcionar una interfaz de usuario intuitiva y atractiva, permitiendo a los usuarios interactuar con la aplicación de manera efectiva.

Además de la presentación visual, el frontend se encarga de gestionar la comunicación con el backend del sistema a través de la API REST. Esto implica el envío de solicitudes HTTP, como GET, POST, DELETE, PUT, etc., para interactuar con los recursos y servicios proporcionados por el backend. Estas solicitudes permiten obtener datos, enviar información, actualizar registros y realizar otras acciones necesarias para el funcionamiento del sistema.

4.3.2 Backend

En el lado del servidor se han implementado una serie de tecnologías orquestadas por Docker, que facilitan el despliegue y la gestión del sistema. El backend se compone de varios servicios que trabajan en conjunto para proporcionar la funcionalidad necesaria, los cuales son descritos a continuación:

API REST

Una API REST (Representational State Transfer) es un conjunto de reglas y convenciones que define cómo se pueden comunicar dos sistemas o aplicaciones a través de Internet. Es un estilo arquitectónico ampliamente utilizado para el diseño de APIs que permiten el intercambio de datos y la interacción entre sistemas de manera eficiente y escalable. En el margen de este proyecto, la API REST corresponde al eje central del backend y se encarga de diversas tareas como el procesamiento y resolución de las solicitudes HTTP provenientes del frontend, la comunicación con la base de datos y, además, gestiona toda la lógica de negocios del sistema.

La arquitectura de diseño de la API se basa en DDD (Domain Driven Design), el cual es un enfoque propuesto por Eric Evans donde todo el diseño del software se centra en comprender y modelar el dominio del problema, es decir, el ámbito o la temática específica en la que se encuentra el software. Además, el DDD aborda la modularidad y la organización del código en torno al dominio. Esto implica la identificación de límites contextuales y separación de responsabilidades del software mediante la definición de

diferentes subdominios y la creación de módulos o componentes que se centran en un subconjunto específico del dominio. De esta forma, se mejora la organización del código disminuyendo el acoplamiento del software, lo que permite que sea más escalable y mantenible.

El modelo de datos utilizado por la API REST define la estructura y las relaciones entre todas las entidades. Mediante este esquema de datos y la implementación de un ORM (Object-Relational Mapping) como Sequelize que funciona como una capa de abstracción entre el código de la aplicación y la base de datos, permitiendo manipular los datos de la base de datos a través de objetos y operaciones orientadas a objetos. De esta forma, se simplifica todo el proceso de comunicación y gestión de la base de datos.

MariaDB

Como sistema gestor de base de datos relacional se utiliza MariaDB. La implementación del servicio se realiza a través de una imagen de Docker, lo que facilita su despliegue y gestión. Esto significa que el servicio de MariaDB y la base de datos asociada pueden ser fácilmente reproducidos en diferentes entornos, lo que simplifica el despliegue y asegura la consistencia. MariaDB es flexible en cuanto a los modelos de datos que puede manejar, lo que permite adaptarse a diferentes necesidades y esquemas, lo que facilita la manipulación de los datos.

phpMyAdmin

De manera complementaria, se incluyó un servicio de administración de alto nivel para la base de datos. Este servicio permite acceder a la base de datos de forma remota desde el navegador mediante un cliente web y facilita la gestión y administración de la base de datos, ofreciendo una interfaz amigable para realizar tareas como consultas, creación de tablas, administración de usuarios y más.

4.3.3 Interacción Humano-Hongo

El tercer módulo principal en la arquitectura del sistema es la Interacción Humano-Hongo. Este módulo consiste en un dispositivo físico que permite capturar los datos de un hongo. Está compuesto por una placa Arduino UNO conectada a sensores de humedad y temperatura, así como a un módulo HC-06 que proporciona una interfaz Bluetooth.

El dispositivo interactúa directamente con la aplicación móvil a través de la tecnología Bluetooth. Esta conexión permite la transferencia de datos desde los sensores del hongo al frontend de la aplicación móvil. De esta manera, los usuarios pueden obtener información en tiempo real sobre las condiciones de humedad y temperatura del hongo.

Este módulo se integra en la arquitectura del sistema para brindar una funcionalidad adicional y enriquecer la experiencia de los usuarios. Los datos capturados por el dispositivo pueden ser utilizados por la API REST para realizar análisis, generar informes o tomar decisiones basadas en las métricas del hongo.

En la Figura 6 adjunta se muestra un esquema general de la arquitectura del sistema, donde se ilustra la interacción entre los diferentes componentes mencionados anteriormente. Este esquema proporciona una visión global de cómo se conectan y comunican entre sí los distintos módulos del sistema, y cómo se integran para ofrecer una experiencia completa y funcional a los usuarios.

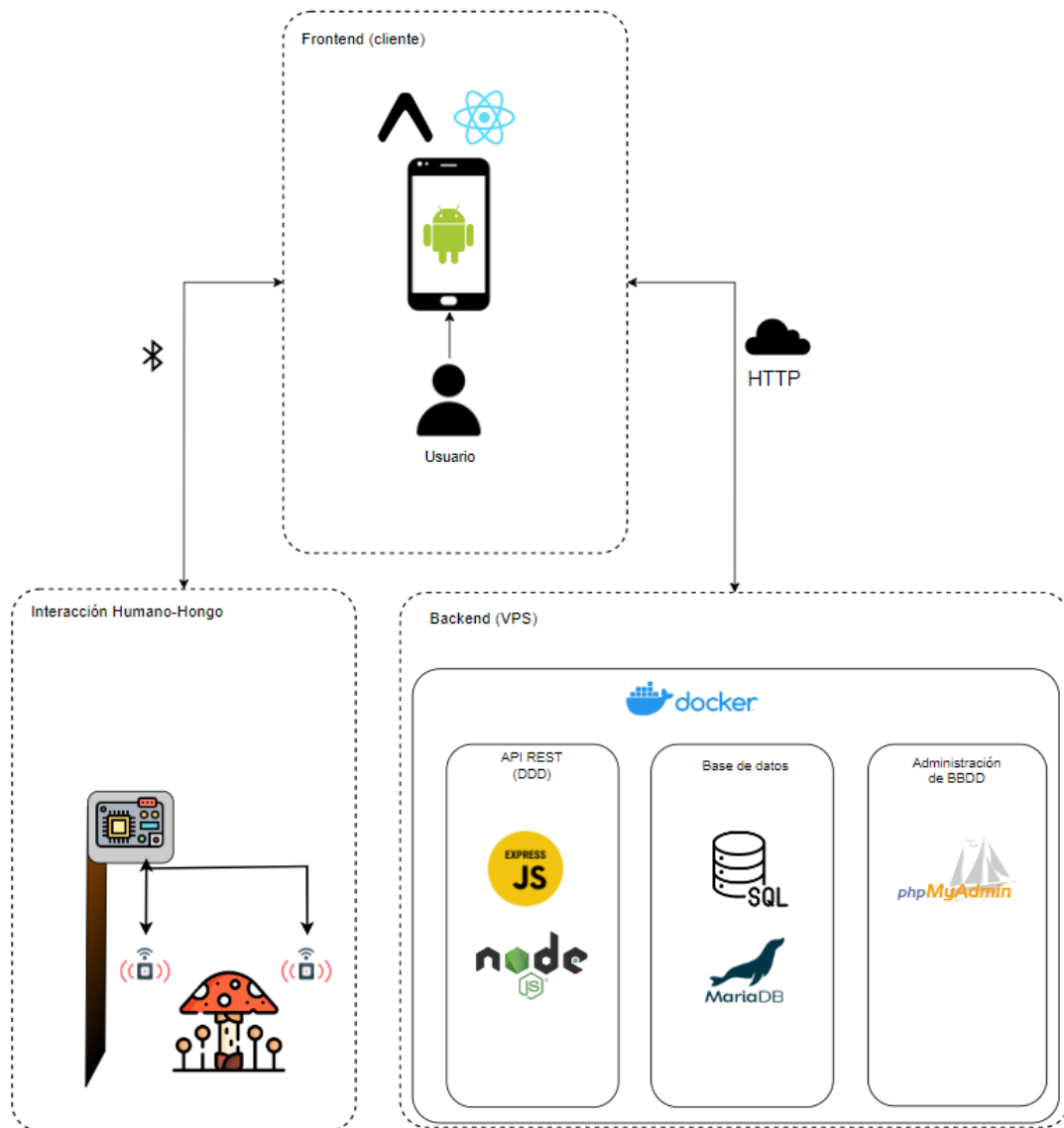


Figura 6. Esquema de arquitectura general de arquitectura.

5. DESARROLLO DEL PROYECTO

5.1 Introducción

El desarrollo de software es un proceso complejo que requiere una gestión efectiva para garantizar la entrega exitosa de un producto de calidad. En este capítulo, se presentará una documentación detallada de todo el desarrollo de software aplicando metodologías ágiles. El desarrollo del proyecto se planificó en tres iteraciones de desarrollo o Sprints de cuatro semanas de duración cada uno como puede observarse a continuación en el plan de desarrollo en la Figura 7.

Sprint	Abril 2023					Mayo 2023					Junio 2023				
	Semana					Semana					Semana				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1. Desarrollo de la aplicación móvil															
2. Desarrollo del dispositivo de interacción															
3. Integración del sistema y mejoras															

Figura 7. Planificación inicial de desarrollo.

5.2 Resultados Toma de requisitos

5.2.1 Resultados entrevistas

Para efectuar la toma de requisitos, se llevó a cabo una sesión de entrevistas a dos stakeholders relevantes del proyecto. Particularmente, se entrevistó al director del proyecto, quien es un diseñador industrial, y también a una profesora Auxiliar de la Facultad de Ciencias Agrarias y Alimentarias de la Universidad Austral de Chile, quien tiene un vasto conocimiento acerca los hongos.

Entrevista al director del proyecto

La finalidad de esta entrevista fue principalmente averiguar acerca de los antecedentes del proyecto “Hongos y Humanos” y conocer las expectativas y sugerencias que tenía acerca de la aplicación móvil desde la perspectiva del diseño.

Él es un diseñador de objetos quien se ha especializado en el diseño de sistemas en la ciudad y el estudio de las interacciones con la comunidad. También se dedica a la pedagogía donde ha tenido más acercamientos al diseño de sistemas de interacción y la cohabitación de especies trabajando en proyectos multidisciplinarios similares.

Además, mencionó que el proyecto “Hongos y Humanos” surgió a partir de propuestas previas con enfoques similares. De acuerdo con la hipótesis planteada, la conexión del reino Fungi con este proyecto se debe a que él reconoce la importancia de los hongos como tema de divulgación en Chile, especialmente en la ciudad de Valdivia, considerada como la “capital” Fungi del país. Él cree firmemente en el potencial de combinar lo físico y lo digital, y tiene grandes expectativas para esta aplicación móvil.

Aunque los parámetros del proyecto aún no están completamente definidos, compartió algunos de los requisitos que espera encontrar en esta aplicación móvil. Uno de ellos es la capacidad de medir diversos datos de especies de hongos mediante sensores físicos o interacciones con la comunidad, y luego traducirlos en conceptos más sencillos, como, por ejemplo, un indicador de "felicidad" de un hongo basado en estos parámetros. También desea registrar las visitas y/o interacciones entre humanos y hongos, estableciendo así una trazabilidad que permita visualizar el desarrollo de estas relaciones a lo largo del tiempo. Además, espera poder realizar publicaciones, compartir imágenes y contenido audiovisual a través de la aplicación.

Estas características y funcionalidades son importantes para el experto en diseño, ya que reflejan su visión de una experiencia integral para los usuarios de la aplicación. El objetivo es crear una plataforma que no solo fomente la interacción entre humanos y hongos, sino que también proporcione un espacio para compartir conocimientos, promover la divulgación y enriquecer la comunidad de entusiastas de los hongos.

Es importante tener en cuenta que los requisitos mencionados por el experto en diseño aún están en proceso de definición y podrían evolucionar a medida que avance el proyecto. Sin embargo, su participación y sus expectativas demuestran su compromiso y entusiasmo por el desarrollo de esta aplicación móvil.

Entrevista a especialista en hongos

Esta entrevista tuvo por finalidad conocer la perspectiva de una persona experta en el mundo de los hongos para así recopilar aspectos importantes acerca de estos y conocer sus expectativas en cuanto a la aplicación y algunas dificultades que pudieran surgir.

Ella es una agrónoma especialista en hongos, y al igual que el experto en diseño y director del proyecto, tiene experiencia en pedagogía. Posee un profundo conocimiento del estado de conservación del reino Fungi en Chile, especialmente en la selva valdiviana. Su compromiso con la divulgación se refleja en su participación en eventos como el FungiFest de Valdivia, lo cual se alinea perfectamente con la hipótesis de este proyecto.

Aunque ella no se encuentra trabajando activamente en el proyecto, ha aportado datos muy relevantes sobre los hongos. Por ejemplo, ha señalado que los hongos son organismos mucho más complejos que las plantas y que, dependiendo de las especies, trasladarlos de su ecosistema natural y llevarlos al sitio donde se ubicarán estos dispositivos de

cohabitación puede ser inviable o requerir un tiempo prolongado para su desarrollo. Además, mencionó que algunos de los datos más importantes para estudiar un hongo son la temperatura, humedad y pH en caso de ser posible. Con esta información se pudo establecer algunos requisitos relacionados a la integración del sistema con microcontroladores y sensores.

Un aspecto muy importante mencionado por la especialista es que la fructificación de los hongos es estacional, lo que significa que no siempre son visibles en la superficie de la tierra. Este hecho no se había tenido en cuenta hasta ahora. Además, ella ha proporcionado datos relevantes para medir los hongos y ha mencionado algunos rangos de aceptación.

Además de su aporte en términos de conocimiento, la especialista ha compartido algunas características y funcionalidades que le gustaría encontrar en la aplicación. Por ejemplo, le gustaría que esté diseñada tanto para usuarios principiantes en el tema como para aficionados y expertos en hongos. Asimismo, considera importante que la aplicación permita compartir registros de hongos y clasificarlos.

Resultados generales de las entrevistas

En general, ambos usuarios esperan una propuesta usable con un diseño atractivo para la comunidad y que logre adaptarse a los diferentes tipos de usuarios que ingresarán en cuanto a la cantidad y complejidad de la información presentada. Ambos esperan medir datos relevantes de estos hongos. Algunas restricciones que considerar son por ejemplo que los hongos son organismos mucho más complejos de lo que comúnmente se piensa y de fructificación estacional, por lo tanto, no siempre son identificables a simple vista. Otra consideración importante son los resultados de postulación Fondart del proyecto, lo cual definiría en gran medida la escala y alcance de este sistema.

Historias de usuario

Como resultado de estas entrevistas, se definieron una serie de historias de usuario con algunos de los requisitos del sistema. Estas historias se agruparon en “épicas”, que, en el contexto de gestión ágil de proyectos, representan una unidad de trabajo grande y significativa que se utiliza para describir una funcionalidad de alto nivel o característica completa que agrega valor al producto final.

En el caso específico de este proyecto, se identificaron dos épicas que desempeñarán un papel crucial en la priorización de requisitos y en la planificación del desarrollo. La primera épica, denominada "interacción con la comunidad", se centra en las principales funcionalidades relacionadas con la red social. Este conjunto de funcionalidades permitirá a los usuarios interactuar, conectarse y compartir información dentro de la comunidad, lo cual constituye un aspecto clave del producto. Estas funcionalidades específicas se encuentran detalladas en la Tabla 14.

La segunda épica, titulada "interacción con los hongos", agrupa las funcionalidades principales relacionadas con la interacción entre los usuarios y los hongos, aprovechando la integración de diversas tecnologías. Esta épica se enfoca en la utilización de un dispositivo móvil con una aplicación específica, así como en la integración de un microcontrolador con sensores. Estas tecnologías permiten a los usuarios interactuar de manera directa con los hongos, recopilando datos relevantes y obteniendo información valiosa. La Tabla 15 proporciona una visión más detallada de las funcionalidades involucradas en esta épica.

Estas épicas no solo sirven como un marco de referencia conceptual, sino que también actúan como guías prácticas durante todo el ciclo de vida del proyecto. Al desglosar cada épica en historias de usuario más pequeñas y manejables, se puede abordar los requisitos específicos de manera incremental y progresiva, lo que permite una entrega más rápida y una mayor flexibilidad para realizar ajustes y mejoras según sea necesario. Esto permitirá entre otras cosas abordar de manera eficiente y organizada las funcionalidades del sistema. Además, facilitará la estimación de esfuerzos y tiempos, así como la asignación de recursos. Con este enfoque, se espera lograr un desarrollo ágil y eficaz, enfocándose en agregar valor al producto final y satisfacer las necesidades de los stakeholders y usuarios.

Tabla 14. Épica: Interacción con la comunidad.

Enunciado de la historia	Criterios de aceptación
1. Yo como usuario deseo poder publicar mis registros de hongos y compartirlos con la comunidad.	El sistema debe incluir la opción de hacer comentarios en una publicación.
	El sistema debe permitir al usuario etiquetar o describir cada imagen para mejorar la organización y comprensión de las imágenes.
	El sistema debe permitir al usuario crear y publicar contenido relacionado con los hongos.
2. Yo como usuario experto en hongos, deseo que la aplicación incluya diferentes tipos de perfiles para distinguir a los usuarios dentro de la comunidad.	El sistema debe permitir a los usuarios calificar y valorar las publicaciones.
	El sistema debe permitir a los usuarios cambiar o actualizar su tipo de perfil en cualquier momento.
3. Yo como usuario administrador deseo poder moderar el contenido subido al sistema.	El sistema debe contar con un sistema de moderación para evitar contenidos inapropiados o irrelevantes.
	El sistema debe incluir un tipo de usuario "moderador"
	Un moderador debe poder eliminar cualquier publicación que incumpla las normas establecidas.

Tabla 15. Épica: Interacción con los hongos.

Enunciado de la historia	Criterios de aceptación
4. Yo como responsable del proyecto deseo poder contar con un dispositivo que permita interactuar con los hongos.	La aplicación debe poder vincularse mediante bluetooth con un dispositivo que albergue el microcontrolador y los sensores.
5. Yo como usuario experto en hongos, deseo poder capturar diferentes datos y métricas de una especie de hongo y poder registrarlos para fines de divulgación y estudio.	El sistema debe permitir la medición de temperatura de un hongo observado.
	El sistema debe permitir el registro de observaciones de hongos de manera georreferenciada.
	El sistema debe permitir la visualización de las observaciones con todos los datos capturados.
	El sistema debe permitir la medición de humedad de un hongo observado.

5.2.2 Prototipos

Para representar cada una de las historias de usuario mencionadas anteriormente y brindar una visualización clara de cómo se verían y funcionarían las interfaces, se crearon mockups utilizando la popular herramienta de diseño Figma. Esta elección se basó en su facilidad de uso y su amplia gama de características y herramientas que permiten crear diseños de alta fidelidad.

Creación y visualización de publicaciones

La primera historia de usuario hace referencia a la funcionalidad de poder crear una publicación relacionada al reino Fungi y publicarla a la comunidad. Para ello, se diseñó un wireflow con la secuencia lógica para completar esta tarea (ver Figura 8). El wireflow comienza con la pantalla de comunidad, donde los usuarios pueden visualizar las publicaciones existentes y crear nuevas. Al presionar el botón “Nueva publicación”, se navega a una nueva ventana con el formulario para crear la publicación. El primer campo requerido es el contenido textual de la publicación. Opcionalmente se puede añadir una foto. Finalmente, al presionar el botón “Publicar” se completa la tarea.

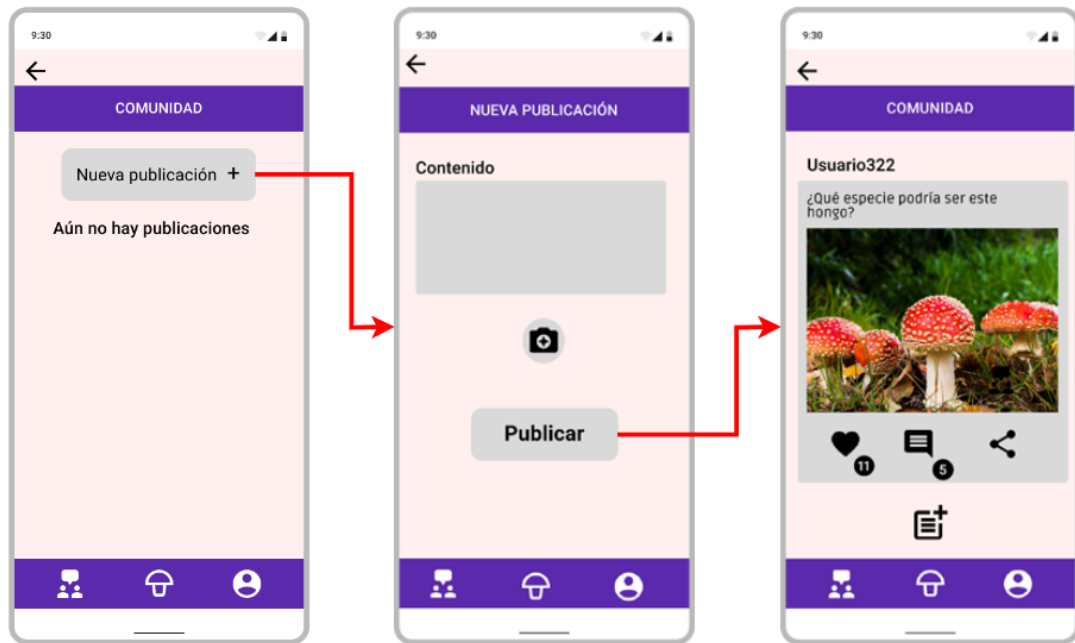


Figura 8. Wireflow para crear una publicación.

Diferenciación de perfiles de usuario

La segunda historia de usuario indica que se debe poder indicar el nivel de conocimiento al momento del registro. En la Figura 9 se propone un mockup para dicha funcionalidad.

Figura 9. Mockup para la diferenciación de usuario.

Moderación de contenido

Otro requisito fundamental, extraído de la tercera historia de usuario, se centra en la necesidad de contar con un sistema de moderación de contenido eficiente dentro de la aplicación. Para dar cumplimiento a esta funcionalidad, se propone la inclusión de un nuevo tipo de usuario llamado "moderador", cuyo rol principal será el de supervisar y eliminar aquellas publicaciones que no cumplan con las directrices establecidas o resulten inapropiadas.

La figura del moderador se presenta como una solución efectiva para garantizar un ambiente seguro y adecuado dentro de la aplicación, fomentando así una comunidad saludable y respetuosa. Al otorgarle a estos usuarios privilegios especiales, como la capacidad de eliminar publicaciones fuera de contexto, se les proporciona una herramienta poderosa para mantener la calidad y coherencia del contenido compartido.

La Figura 10 muestra el worflow desde la perspectiva de un moderador para eliminar una publicación no deseada. La interfaz de un usuario tipo moderador cuenta con un botón adicional para eliminar publicaciones. El flujo comienza al presionar este botón con ícono de “eliminar” y termina al confirmar la acción.

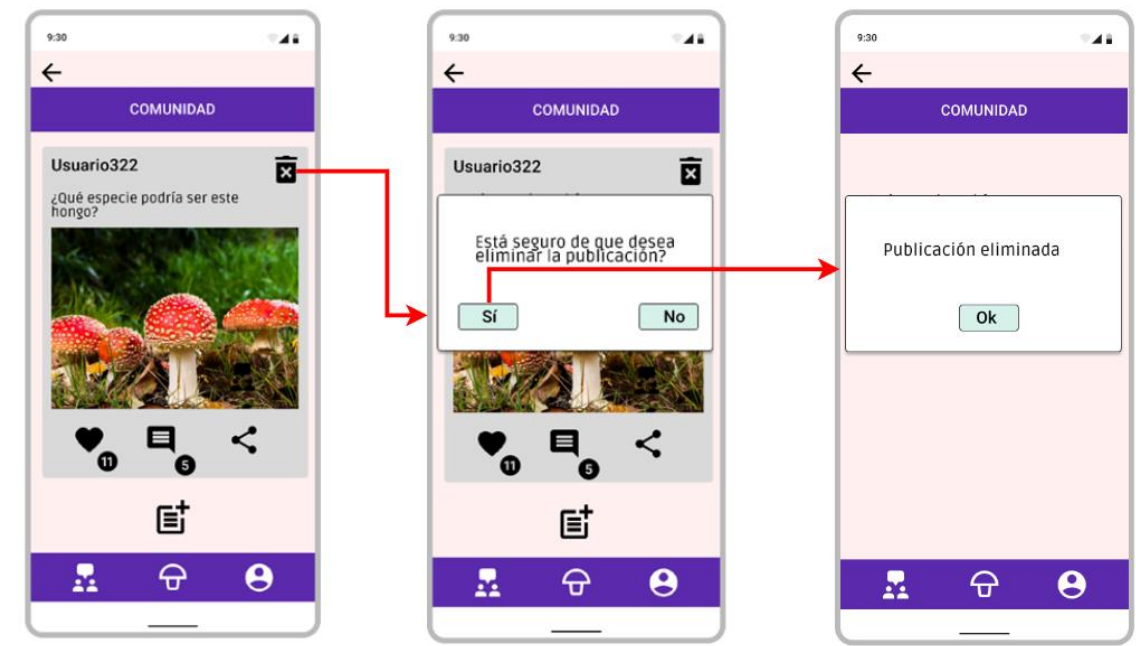


Figura 10. Wireflow para eliminar una publicación.

Conexión con el dispositivo de interacción

La cuarta historia de usuario hace alusión a la funcionalidad de conectarse al dispositivo de interacción mediante una interfaz Bluetooth. La Figura 11 muestra un wireflow donde luego de presionar el botón “Conectar”, el sistema se conecta automáticamente con el módulo Bluetooth del dispositivo de interacción.

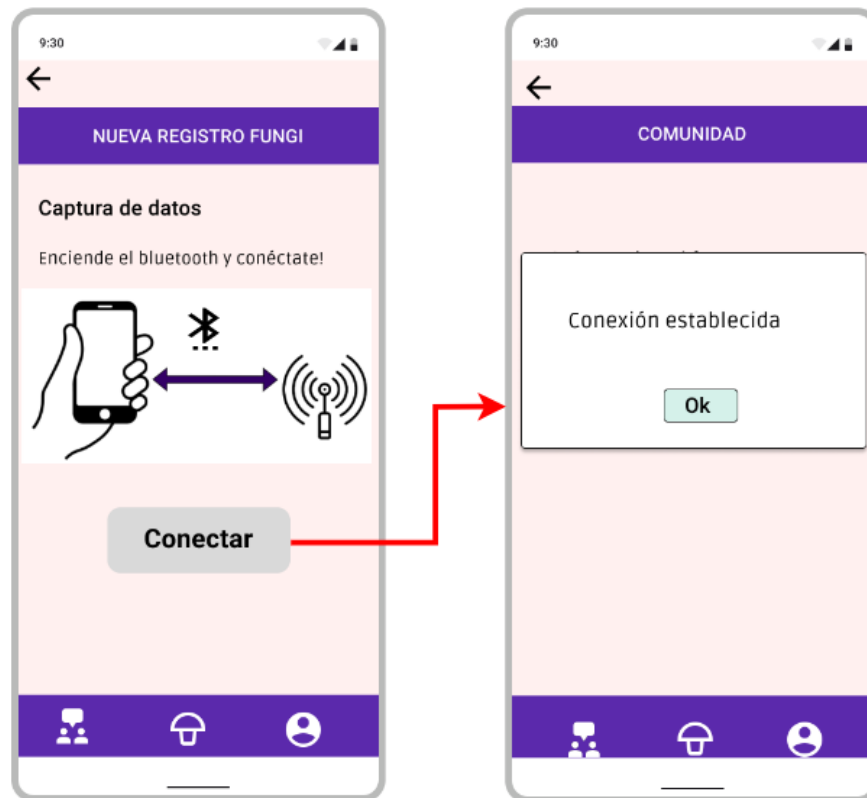


Figura 11. Wireflow para conectarse al dispositivo de interacción.

Captura de datos de los sensores

Por último, se tiene la quinta historia de usuario que engloba la funcionalidad de captura de datos de los sensores. Un requisito para efectuar esta captura de datos es que se haya establecido la conexión con el dispositivo previamente. En esta propuesta de diseño luego de presionar el botón “Medir datos” se muestran los valores de temperatura y humedad extraídos de los sensores como se observa en la Figura 12.

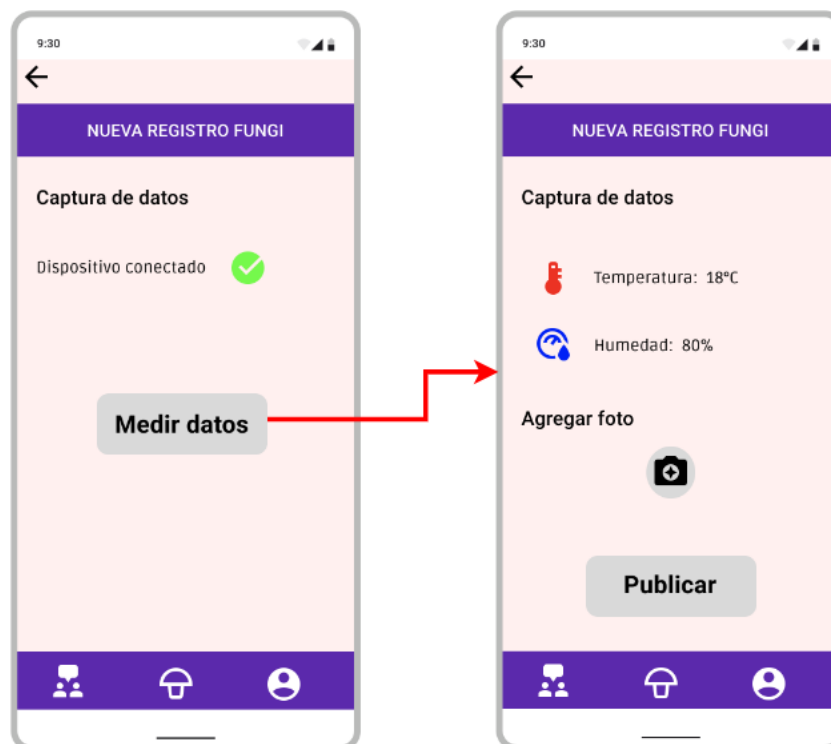


Figura 12. Wireflow para obtener datos de los sensores.

5.2.3 Especificación de requisitos

La especificación de los requisitos del proyecto se realizó en base a historias de usuario recopiladas de entrevistas y reuniones con stakeholders en conjunto con otros requisitos propuestos de manera complementaria con el fin de cumplir los objetivos del proyecto.

Priorización de requisitos

Para establecer un plan de trabajo y determinar la prioridad de los requisitos del sistema, se aplicó una matriz de priorización de requisitos tomando en cuenta a la importancia y la dificultad técnica de cada requerimiento como variables de la matriz. El enfoque adoptado se basó en la premisa de dar prioridad a aquellos requisitos que presentaban una alta importancia y una menor dificultad técnica. También se tuvo en consideración las épicas de historias de usuarios mencionadas en la sección anterior buscando que los requisitos que tengan prioridad similar se encuentren relacionados en términos de funcionalidad. La razón detrás de esta elección era maximizar el impacto del sistema, centrándose en aquellos aspectos críticos que, a su vez, pudieran implementarse con relativa facilidad. De esta manera, se buscaba garantizar la obtención de resultados tangibles en el menor tiempo posible. Tras aplicar este método, se asignó un ranking a cada requisito que puede observarse en la columna “Rank” de la Tabla 14 y Tabla 15.

Requisitos funcionales

A continuación, se muestran en detalle los requisitos funcionales del sistema (ver Tabla 16).

Tabla 16. Requisitos funcionales.

Identificador	Requisito	Importancia	Dificultad	Rank
RF1	El sistema debe contar con una base de datos remota.	10	5	1
RF2	El sistema debe contar con una API REST en el lado backend.	10	7	2
RF3	La aplicación debe permitir la creación y gestión de cuentas usuario.	10	5	3
RF4	La aplicación debe permitir crear publicaciones en la sección de comunidad.	10	5	4
RF5	El sistema debe permitir a los usuarios indicar su nivel de experiencia y conocimientos con los hongos al crear su perfil.	8	4	6
RF6	El sistema debe mostrar el tipo de perfil de un usuario en las publicaciones y comentarios para que otros usuarios puedan ver el nivel de conocimiento de la persona que lo escribió	8	4	7
RF7	La aplicación debe permitir tomar fotografías a una especie.	8	4	8
RF8	El sistema debe contar con un microcontrolador que capture y envíe los datos capturados por los sensores.	10	8	9
RF9	La aplicación debe permitir la toma de datos mediante sensores de humedad y temperatura.	10	9	10
RF10	La aplicación debe permitir comentar en las publicaciones.	9	7	11
RF11	La aplicación debe permitir añadir registros de hongos con todos los datos medidos.	8	8	12
RF12	La aplicación debe permitir reaccionar en las publicaciones.	7	7	13
RF13	La aplicación debe permitir filtrar las publicaciones por popularidad o relevancia para que los usuarios puedan ver la información más valiosa y útil primero.	7	7	14
RF14	La aplicación debe contar con un sistema de moderación.	6	7	15
RF15	La aplicación debe incluir visualizaciones de análisis de datos.	5	5	16

Requisitos no funcionales

A continuación, se indican los requisitos no funcionales que se han definido para el sistema (ver Tabla 17).

Tabla 17. Requisitos no funcionales

Identificador	Requisito	Importancia	Dificultad	Rank
RNF1	La aplicación debe ser segura, manteniendo la privacidad y seguridad de los datos de los usuarios	10	5	1
RNF2	La aplicación debe ser fácil de usar e intuitiva para permitir que usuarios de diferentes niveles de habilidad puedan utilizarla sin dificultad	10	5	2
RNF3	La aplicación móvil debe funcionar en dispositivos con sistema operativo Android 5 o superior.	10	6	3

5.3 Primer Sprint: Desarrollo de la aplicación móvil

5.3.1 Resumen del Sprint

Dentro de este Sprint se incluyó el primer hito del proyecto que consistió en el desarrollo y despliegue del primer incremento de la aplicación móvil funcional con el objetivo poder efectuar una validación inicial del software mediante pruebas de usabilidad.

5.3.2 Tareas propuestas

Respecto a las tareas propuestas para esta iteración, el desarrollo se centró en la infraestructura base del sistema, lo que incluye el modelado de datos y desarrollo de backend y frontend abarcando las historias de usuario incluidas en la Épica de interacción con la comunidad descrita anteriormente donde básicamente se abarcan las funcionalidades relacionadas al componente de red social. De manera más específica, se definieron una serie de tareas con sus requisitos funcionales respectivos en la Tabla 18.

Tabla 18: Tareas del primer Sprint.

Tarea	Requisito(s) funcional(es) asociado(s)		Dificultad
Diseño y creación de base de datos	RF1	El sistema debe contar con una base de datos remota.	Media
Desarrollo backend	RF2	El sistema debe contar con una API REST en el lado backend.	Alta
Desarrollo frontend	RF3	La aplicación debe permitir la creación y gestión de cuentas usuario.	Alta
	RF4	La aplicación debe permitir crear publicaciones en la sección de comunidad.	
	RF5	El sistema debe permitir a los usuarios indicar su nivel de experiencia y conocimientos con los hongos al crear su perfil.	
	RF6	El sistema debe mostrar el tipo de perfil de un usuario en las publicaciones y comentarios para que otros usuarios puedan ver el nivel de conocimiento de la persona que lo escribió.	
	RF10	La aplicación debe permitir comentar en las publicaciones.	
	RF7	La aplicación debe permitir tomar fotografías a una especie.	
	RF14	La aplicación debe contar con un sistema de moderación.	
	RF12	La aplicación debe permitir reaccionar en las publicaciones.	
	RF13	La aplicación debe permitir filtrar las publicaciones por popularidad o relevancia para que los usuarios puedan ver la información más valiosa y útil primero.	

5.3.3 Resultados del Sprint

Diseño y creación de base de datos

La primera tarea definida para esta iteración fue el modelado y creación de la base de datos. Esto consiste en establecer la estructura de las diferentes entidades y objetos dentro del sistema, así como sus atributos y relaciones. Posteriormente y a partir de ese modelo se procede a crear la base de datos.

De acuerdo con un análisis y esquema preliminar, se identificaron cinco entidades principales dentro del sistema, las cuales se describen a continuación. El modelo de datos del sistema se representa en un diagrama entidad-relación (ver Figura 13).

- Usuario: Representa a cada persona que se registra, accede y utiliza la plataforma. Cada usuario tiene una identidad única y puede tener diferentes roles, permisos y atributos asociados.

- **Publicación:** Es una representación de un contenido compartido por los usuarios en la plataforma, cada publicación es única, puede contener texto e imágenes, y está relacionada al autor quien es de tipo Usuario.
- **Comentario:** Representa las respuestas o reacciones de los usuarios a una publicación específica en la plataforma. Cada comentario está asociado a una publicación y se crea por un usuario en particular.
- **Registro Fungi:** Corresponde a un tipo a un tipo de publicación especializada cuya distinción es que permite registrar observaciones de hongos capturando datos de localización y métricas de los sensores con ayuda del dispositivo de interacción que se desarrollará más adelante.

Adicionalmente, se definieron entidades secundarias como “Like” que representa las reacciones de los usuarios a una publicación, y “Sesión” que permite llevar un control de acceso y autorización mediante la implementación de tokens de usuario.

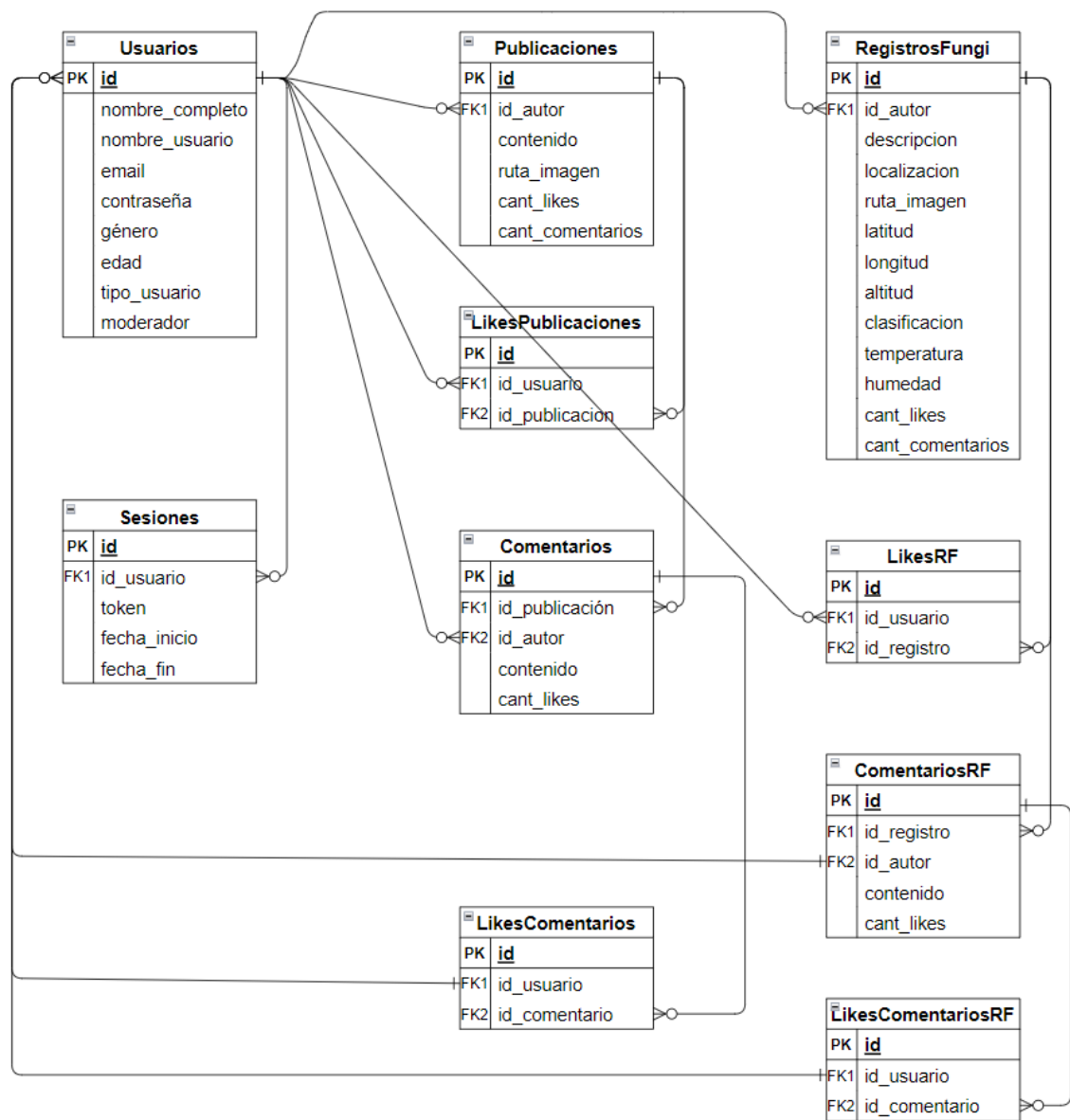


Figura 13. Diagrama Entidad-Relación del sistema.

Una vez que se definió el modelo de datos preliminar del sistema, se inició el desarrollo de la parte del backend que se encargaría de interactuar con la base de datos. Para esto, se creó un proyecto con Node.js y se utilizó TypeScript para escribir el código que representaría cada entidad del modelo. Se empleó la biblioteca Sequelize, la cual es un ORM (Object-Relational Mapping), para facilitar el mapeo automático de objetos del programa a tablas de la base de datos.

Sequelize permitió implementar las relaciones entre las entidades de manera sencilla y simplificó todo el proceso de comunicación con la base de datos. Además, se utilizó una instancia del servidor de base de datos MariaDB, la cual fue creada a partir de una imagen

Docker. La configuración de esta instancia se especificó en el archivo docker-compose de tipo YAML del proyecto, donde se detallan todas las opciones de configuración de los servicios que serán desplegados, estos parámetros de configuración pueden apreciarse en la Tabla 19.

Tabla 19: Configuración del servicio MariaDB.

Parámetro	Valor	Descripción
container_name	mariadb_database	Esto establece el nombre del contenedor como "mariadb_database". El contenedor se identificará con este nombre en el entorno Docker.
image	mariadb:jammy	Especifica la imagen de Docker que se utilizará para crear el contenedor. En este caso, se está utilizando la imagen de MariaDB con la etiqueta "jammy" que hace referencia a una versión de Ubuntu.
ports	\${PORTDB}:3306	Aquí se define la configuración de los puertos. Esto indica que el puerto de la máquina anfitriona \${PORTDB} se mapeará al puerto 3306 del contenedor. El puerto 3306 es el puerto predeterminado para la comunicación con la base de datos MariaDB.
volumes	./mariadb:/var/lib/mysql	Esta configuración especifica que el directorio local “./mariadb” se montará en el directorio “/var/lib/mysql” dentro del contenedor. Esto permite persistir los datos de la base de datos en la máquina anfitriona, lo que significa que los datos se conservarán incluso si el contenedor se reinicia o se elimina.
enviroment	MARIADB_ROOT_PASSWORD=\${PASSWORD}	Aquí se establece la variable de entorno MARIADB_ROOT_PASSWORD con el valor \${PASSWORD}. Esta variable se utiliza para establecer la contraseña de usuario root de la base de datos MariaDB.
	MARIADB_DATABASE=\${NAMEDB}	Esta línea establece la variable de entorno MARIADB_DATABASE con el valor \${NAMEDB}. Esta variable se utiliza para especificar el nombre de la base de datos que se creará en MariaDB.

Para simplificar la administración de la base de datos, se ha integrado phpMyAdmin, un servicio complementario que proporciona una interfaz gráfica para gestionar todos los aspectos de la base de datos de forma remota a través de un navegador web. Esta herramienta ofrece una manera intuitiva y conveniente de interactuar con la base de datos, facilitando tareas como la creación y modificación de tablas, la ejecución de consultas SQL, la importación y exportación de datos, entre otras funcionalidades. La integración de phpMyAdmin en el sistema ofrece una forma accesible y eficiente de administrar la base de datos, lo que contribuye a agilizar el proceso de desarrollo y mantenimiento del sistema.

El servicio phpMyAdmin, al igual que MariaDB, fue implementado utilizando Docker a partir de una imagen disponible en línea. Para orquestar el despliegue de todo el sistema, se agregó la configuración de este servicio al archivo YAML docker-compose. A continuación, se detallan los parámetros de configuración en la Tabla 20. Esta configuración permite desplegar y ejecutar el servicio phpMyAdmin junto con MariaDB, proporcionando una interfaz gráfica para administrar la base de datos de manera remota. Al utilizar Docker, se logra una fácil instalación y configuración del servicio, lo que simplifica el proceso de despliegue del sistema completo.

Tabla 20: Configuración del servicio phpMyAdmin.

Parámetro	Valor	Descripción
depends_on	mariadb_service	Esta línea indica que phpMyAdmin depende del servicio de base de datos MariaDB. Asegura que el servicio de base de datos esté en funcionamiento antes de iniciar phpMyAdmin.
image	phpmyadmin:5.2.0-apache	Se utiliza la imagen de Docker “phpmyadmin:5.2.0-apache” para ejecutar phpMyAdmin. Esta imagen contiene una versión específica de phpMyAdmin y utiliza el servidor web Apache.
ports	8080:80	Esta configuración establece que el puerto 8080 del host se mapeará al puerto 80 dentro del contenedor de phpMyAdmin. Esto permite acceder a la interfaz de phpMyAdmin a través del puerto 8080 del navegador web.
volumes	./phpmyadmin/config.user.inc.php :/etc/phpmyadmin/config.user.inc.php	Mediante esta configuración de volúmenes, se monta un archivo de configuración personalizado (config.user.inc.php) en el directorio “/etc/phpmyadmin/” dentro del contenedor de phpMyAdmin.
enviroment	PMA_ARBITRARY=1	Esta variable de entorno se establece en 1, lo que indica que phpMyAdmin permitirá la conexión a cualquier servidor de bases de datos.

Desarrollo backend

Tal como se mencionó en el capítulo anterior, se implementó la API REST utilizando el framework ExpressJS de Node³. La API se ha diseñado y construido como parte de un proyecto con el objetivo de proporcionar un servicio web escalable y flexible para interactuar con recursos específicos a través de operaciones HTTP estándar.

El uso de Express para desarrollar la API REST ofrece numerosas ventajas en términos de facilidad de uso, eficiencia y mantenibilidad. Una de las características destacadas de Express es su enfoque sencillo y claro para definir rutas y controladores. Esto permite una

³ Mora, D. (2023). Desarrollo de API REST para la aplicación Micelio. [Repositorio del desarrollo backend]. Recuperado de <https://github.com/dmoram/rest-api-fungi>

organización lógica y estructurada de los endpoints de la API, lo que facilita el mantenimiento y comprensión del código.

Además, Express ofrece el concepto de middleware, que es extremadamente útil para implementar características adicionales en la API. El middleware permite agregar funcionalidades como autenticación, validación de datos y manejo de errores de manera modular y reutilizable. Esto simplifica el proceso de implementación de características comunes en la API, ya que se pueden escribir middleware una vez y reutilizarlos en diferentes rutas o endpoints.

Siguiendo el esquema DDD (Domain Driven Design) que se está aplicando en toda la arquitectura del sistema, la organización del código fuente de la API se ha modularizado a nivel de entidades. Esto significa que cada entidad dentro del sistema se representa como un módulo independiente en la API, encapsulando su lógica de negocio y comportamiento específico. Esta modularización incrementa la cohesión dentro de cada módulo, lo que facilita la comprensión, separación de responsabilidades y el mantenimiento de las funcionalidades asociadas a cada entidad. A su vez, dentro de cada módulo se han definido diferentes submódulos o capas de abstracción. En primer lugar, se tiene la capa de dominio donde se han definido los modelos de datos de las entidades usando el ORM Sequelize. Luego, se tiene la capa de aplicación donde se han implementado los controladores, estos se utilizan para manejar las solicitudes HTTP entrantes y coordinar la lógica de negocio correspondiente. Los controladores son responsables de recibir los datos de entrada, procesarlos, interactuar con los diferentes servicios o componentes necesarios y generar las respuestas adecuadas. Por último, se tiene la capa de infraestructura donde se implementan los middlewares y las rutas. Los middlewares corresponden a funciones o componentes que se ejecutan en el flujo de procesamiento de una solicitud HTTP antes de llegar a su controlador final. Estas funciones pueden realizar tareas como la validación de datos, la autenticación de usuarios, el registro de solicitudes, la gestión de errores, la compresión de respuestas, entre otros. Por otro lado, las rutas definen la estructura de las URL y especifican qué controlador se debe invocar para manejar una solicitud particular. Las rutas determinan cómo se accede a los recursos en la API y qué operaciones se pueden realizar sobre ellos.

La API desarrollada proporciona un conjunto de endpoints que permiten a los usuarios interactuar con diferentes recursos del sistema y realizar diversas acciones. A continuación, se presenta una documentación detallada de los endpoints desarrollados para cada una de las entidades del sistema. Desde la Tabla 21 a la Tabla 24 se describe cada endpoint definido con su ruta de acceso respectiva, sus parámetros requeridos y la respuesta del servidor.

Tabla 21: Descripción de endpoints relacionados a los comentarios.

Endpoint	Ruta	Solicitud	Parámetros (body)	Respuesta
Crear comentario	api/comments	POST	content (string): Contenido del comentario.	200 OK
			post_id (string): ID de la publicación a la que se añade el comentario. author_id (string): ID del autor del comentario.	Body: {“ok”: true}
Obtener comentarios	api/comments/:post_id	GET	{ }	200 OK
				Body: Lista de comentarios en formato JSON.
Actualizar reacciones	api/comments/likes	PUT	comment_id (string): ID del comentario.	200 OK
			user_id (string): ID del usuario que realiza la acción. action (string): Acción a realizar. Los valores posibles son "like" o "dislike".	Body: {“ok”: true}

Tabla 22: Descripción de endpoints relacionados a las publicaciones.

Endpoint	Ruta	Solicitud	Parámetros (body)	Respuesta
Crear publicación	api/posts	POST	<p>content (string): El contenido de la publicación.</p> <p>author_id (string): El ID del autor de la publicación.</p> <p>likes (opcional, número): El número inicial de "likes" de la publicación.</p> <p>image (opcional, archivo): La imagen asociada a la publicación.</p>	200 OK
				Body: {“ok”: true}
Obtener publicaciones	api/posts	GET	{ }	Status: 200
				Body: Lista de publicaciones en formato JSON.
Actualizar reacciones	api/posts/likes	PUT	<p>post_id (string): ID del comentario.</p> <p>user_id (string): ID del usuario que realiza la acción.</p> <p>action (string): Acción a realizar. Los valores posibles son "like" o "dislike".</p>	Status: 200
				Body: {“ok”: true}
Obtener imagen de una publicación	api/posts/:id	GET	{ }	Status: 200
				Body: Imagen de la publicación
Obtener cantidad de publicaciones de un usuario	api/posts/count/:user_id	GET	{ }	Status: 200
				Body: { "user_id": user_id, "postCount": 5 }
Eliminar publicación	api/posts/:post_id	DELETE	{ }	Status: 200
				Body: {“ok”:true}

Tabla 23: Descripción de endpoints relacionados a los usuarios.

Endpoint	Ruta	Solicitud	Parámetros (body)	Respuesta
Crear usuario	api/usuarios	POST	Objeto JSON con todos los datos del usuario a crear	200 OK
				Body: {"ok": true}
Iniciar sesión de usuario	api/usuarios/login	POST	email (string): El email del usuario. password (string): La contraseña del usuario.	200 OK
				Body: { "id": 123, "token": "jwt_token", "mod": false }
Cerrar sesión de usuario	api/usuarios/logout/:id	POST	{}	200 OK
				Body: {"ok": true}
Obtener datos de un usuario	api/usuarios/:id	GET	{}	200 OK
				Body: Datos del usuario
Eliminar usuario	api/usuarios/:id	DELETE	{}	200 OK
				Body: {"ok": true}
Actualizar usuario	api/usuarios/:id	PUT	Objeto JSON con los nuevos datos del usuario a actualizar, incluyendo nombre, email y/o contraseña.	200 OK
				Body: {"ok": true}

Tabla 24: Descripción de endpoints relacionados a los registros Fungi.

Endpoint	Ruta	Solicitud	Parámetros (body)	Respuesta
Crear usuario	api/usuarios	POST	Objeto JSON con todos los datos del usuario a crear	200 OK
				Body: {“ok”: true}
Iniciar sesión de usuario	api/usuarios/login	POST	email (string): El email del usuario. password (string): La contraseña del usuario.	200 OK
				Body: { "id": 123, "token": "jwt_token", "mod": false }
Cerrar sesión de usuario	api/usuarios/logout/:id	POST	{ }	200 OK
				Body: {“ok”: true}
Obtener datos de un usuario	api/usuarios/:id	GET	{ }	200 OK
				Body: Datos del usuario
Eliminar usuario	api/usuarios/:id	DELETE	{ }	200 OK
				Body: {“ok”: true}
Actualizar usuario	api/usuarios/:id	PUT	Objeto JSON con los nuevos datos del usuario a actualizar, incluyendo nombre, email y/o contraseña.	200 OK
				Body: {“ok”: true}

Respecto a la seguridad de la API, es fundamental adoptar consideraciones y buenas prácticas para restringir el acceso no autorizado a la información y garantizar un uso adecuado de los recursos del sistema.

Una de las medidas importantes es la encriptación de las contraseñas de los usuarios antes de almacenarlas en la base de datos. Para lograr esto, se utiliza la librería bcrypt.js, que proporciona funciones para realizar el hash y la comparación segura de contraseñas utilizando el algoritmo bcrypt. Al emplear esta librería, las contraseñas se convierten en un hash irreversible antes de ser almacenadas. Esto significa que incluso si alguien obtiene acceso a la base de datos, es extremadamente difícil o prácticamente imposible obtener

las contraseñas originales a partir del hash almacenado. De esta manera, se mejora significativamente la seguridad de las contraseñas de los usuarios.

Además, se ha implementado el uso de tokens de autenticación de usuario utilizando el estándar JWT (JSON Web Token). Esta técnica permite evitar el envío repetitivo de las credenciales de usuario (como nombre de usuario y contraseña) en cada solicitud a la API para acceder a los recursos del sistema. En cambio, cuando un usuario se autentica correctamente, se le emite un token JWT que contiene información sobre su identidad y privilegios. Este token se envía en las solicitudes posteriores como una forma de autenticación. Al validar la firma del token y leer los datos de la carga útil del JWT, el servidor puede verificar la autenticidad del usuario y su autorización para acceder a los recursos solicitados. Esto proporciona una forma segura y eficiente de mantener la sesión del usuario sin la necesidad de enviar las credenciales en cada solicitud, lo que mejora la experiencia del usuario y reduce la carga en el servidor.

Desarrollo frontend

La siguiente etapa dentro del primer hito del proyecto consistió en el desarrollo de la aplicación móvil como tal, utilizando Expo como plataforma de implementación y JavaScript como lenguaje de programación⁴.

La arquitectura del frontend está basada en componentes. Al aplicar este enfoque, el código se organiza en unidades independientes y reutilizables llamadas componentes. Cada componente encapsula una funcionalidad específica y puede comunicarse con otros componentes para construir una aplicación completa. La organización del proyecto puede resumirse en la Figura 14.

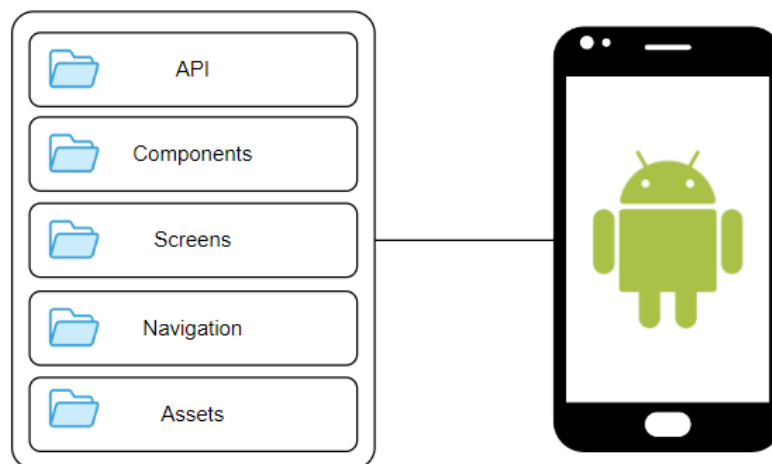


Figura 14. Organización del frontend.

⁴ Mora, D. (2023). Desarrollo de aplicación móvil Micelio. [Repositorio del desarrollo frontend]. Recuperado de <https://github.com/dmoram/fungi-app>

Como se observa en la Figura 14, cada carpeta encapsula un aspecto importante de la aplicación móvil, cada uno de ellos se describe a continuación:

- **API:** Dentro de esta carpeta se define el archivo de configuración que permite establecer comunicación con la API REST. Para ello se crea una instancia de un objeto de conexión con los datos del servidor usando la librería Axios. Este objeto se exporta y puede ser instanciado para realizar las peticiones en cualquier otro componente.
- **Components:** En esta carpeta se organizan todos los componentes funcionales reutilizables dentro del proyecto, es decir, unidades de código que encapsulan una funcionalidad específica y pueden ser utilizadas en diferentes partes de la aplicación. Algunos componentes definidos para la aplicación son ventanas emergentes, publicaciones, componente de cámara, publicaciones, comentarios, etc.
- **Screens:** Esta carpeta contiene todas las pantallas o vistas de la aplicación móvil. Cada pantalla representa una interfaz de usuario completa y muestra un conjunto específico de información o funcionalidades al usuario. Aquí se definen y se implementan todas las pantallas necesarias para la aplicación, como la pantalla de inicio, la pantalla de perfil de usuario, la pantalla de búsqueda, la pantalla de detalles del producto, etc. Organizar las pantallas en esta carpeta facilita la navegación y el acceso a las diferentes secciones de la aplicación.
- **Navigation:** Aquí se implementan todos los componentes relacionados a la navegación dentro de la aplicación. En este caso, se ha definido que el contenedor principal de la aplicación móvil sea un “Bottom Tab Navigator”, el cual es un tipo de navegación muy común en aplicaciones móviles. Proporciona una barra de pestañas en la parte inferior de la pantalla, permitiendo a los usuarios cambiar entre diferentes pantallas o secciones principales de la aplicación con solo tocar las pestañas correspondientes. Esta forma de navegación es especialmente útil para organizar y navegar por diferentes funcionalidades o vistas principales de una aplicación de manera intuitiva y accesible. Adicionalmente, y de manera complementaria, se implementa una forma de navegación secundaria de tipo “Stack” donde las pantallas se guardan en una estructura de datos tipo pila permitiendo al usuario volver a la pantalla anterior.
- **Assets:** Dentro de esta carpeta, se almacenan todos los recursos gráficos y archivos estáticos utilizados dentro de la aplicación.

Para la implementación de la aplicación móvil se usaron diferentes librerías de Expo y React Native, las principales son listadas a continuación en la Tabla 25.

Tabla 25. Librerías utilizadas en el desarrollo.

Nombre	Versión	Descripción
react-native-async-storage	1.17.11	Una librería para el almacenamiento de datos asincrónico en React Native. Proporciona una interfaz sencilla para guardar y recuperar datos de forma persistente en el dispositivo.
react-native-picker	2.4.8	Un componente de selección de opciones que ofrece una lista desplegable en React Native. Se utiliza para seleccionar un valor de una lista de opciones predefinidas.
react-navigation/bottom-tabs	6.5.7	Una librería que ofrece una navegación de pestañas en la parte inferior de la aplicación. Se utiliza para crear una barra de navegación con pestañas en la parte inferior de la pantalla.
react-navigation/native	6.1.6	Una librería de navegación para React Native que proporciona una API para gestionar la navegación entre pantallas en una aplicación. Permite crear rutas y navegadores para la navegación entre componentes.
react-navigation/stack	6.3.16	Una librería de navegación que proporciona una pila de navegación en React Native. Se utiliza para gestionar la navegación entre pantallas utilizando una pila (last-in, first-out).
axios	1.3.5	Una librería de cliente HTTP basada en promesas que se utiliza para realizar solicitudes HTTP en aplicaciones JavaScript. Proporciona una interfaz sencilla para realizar solicitudes GET, POST, PUT, DELETE, etc.
dayjs	1.11.7	Una librería de manipulación de fechas y horas en JavaScript.
expo-camera	13.2.1	Una librería de Expo para acceder a la funcionalidad de la cámara en dispositivos móviles. Proporciona componentes y API para capturar imágenes y videos, acceder a las cámaras del dispositivo y controlar la configuración de la cámara.
expo-checkbox	2.3.1	Un componente de casilla de verificación personalizado para Expo. Proporciona una casilla de verificación estilizada con opciones adicionales de personalización y manejo de eventos.

Una de las decisiones de diseño más importantes fue elegir el nombre de la aplicación móvil, ya que era un aspecto crucial. Después de considerar varias opciones, se decidió llamarla "Micelio", inspirado en el concepto clave del reino Fungi. El micelio representa la intrincada red vegetativa que conecta a los hongos y se considera fundamental para su existencia. La elección de este nombre se basa en la analogía entre la red del micelio y las redes sociales, ya que ambos conceptos comparten la idea de interconexión. Además, esta

analogía refuerza el objetivo del proyecto, que es relacionar y vincular a las personas con el mundo de los hongos. El logo de la aplicación puede verse en la Figura 15.



Figura 15. Logo de la aplicación móvil.

El diseño de la interfaz de usuario está basado directamente en los primeros mockups obtenidos luego del proceso de toma de requisitos documentado en el Capítulo 3, aunque se hicieron algunas modificaciones como el cambio en la paleta de colores con el objetivo de lograr un producto más usable y con mayor identidad.

A continuación, se proporciona una descripción detallada de cada una de las funcionalidades implementadas en Micelio contempladas en el primer hito de desarrollo del proyecto, brindando información adicional sobre su propósito y funcionalidad específica.

Registro de usuario

Dado que una parte fundamental de este sistema es la formación de una comunidad, el registro de usuarios es una funcionalidad muy importante. Para su implementación, se elaboró un formulario con datos relevantes para la identificación de un usuario de Micelio. Estos datos son:

- Nombre de usuario.
- Nombre completo.
- Correo electrónico.
- Contraseña.
- Edad.
- Nivel de conocimiento.

Con el objetivo de no sobrecargar la pantalla con información y de brindar una mejor experiencia de usuario, el formulario se dividió en dos pantallas como se observa en la Figura 16.

The image displays two sequential mobile application screens for a registration process. The first screen, titled 'Registro: Paso 1', includes input fields for 'Nombre de usuario', 'Nombre completo', 'Correo electrónico', and 'Contraseña', along with a 'Género' dropdown menu currently showing 'Masculino'. The second screen, titled 'Registro: Paso 2', features an 'Indica tu edad' input field and a section titled '¿Qué tanto sabes del mundo Fungi?' with five radio button options: 'Principiante', 'Observador(a)', 'Aficionado(a)', 'Experto(a)', and 'Investigador(a)'. Both screens have a 'Siguiente' button at the bottom right.

Figura 16. Capturas del formulario de registro.

El manejo de errores y las validaciones como por ejemplo los tipos de datos sean correctos o que el correo sea válido se realizan localmente mediante funciones de validación. Ante un campo faltante o error en los datos detectado al presionar el botón “Siguiente”, se despliega una ventana emergente informativa como se observa en la Figura 17.

Todos los datos ingresados por el usuario se almacenan en una estructura y son enviados a la API REST usando Axios para efectuar el registro. Si los datos son válidos y la solicitud se realiza correctamente, el cliente recibe una respuesta exitosa y automáticamente se navega a la pantalla de inicio de sesión finalizando el proceso de registro.



Figura 17. Captura de un mensaje en la ventana informativa.

Inicio de sesión

La siguiente funcionalidad implementada en Micelio es el inicio de sesión, esto permite a los usuarios autenticarse en la aplicación para acceder a su cuenta personalizada y aprovechar todas las características y funcionalidades asociadas. La pantalla de inicio de sesión puede observarse en la Figura 18.

El proceso comienza con la solicitud de las credenciales respectivas del usuario que para esta aplicación se optó por el correo electrónico y la contraseña. Al presionar el botón “iniciar sesión” se realiza una validación local para verificar que el formato de los datos es correcto. Se implementa el mismo sistema de manejo de errores que en el registro. Una vez validados los datos, se envían a la API REST usando Axios. En el servidor, se realiza el proceso de autenticación y se envía la respuesta correspondiente. Si los datos son correctos se genera un token de usuario JWT y se envía el cliente. Este token se almacena localmente usando la librería AsyncStorage y se usa para realizar cualquier solicitud que requiera autenticación.



Figura 18. Captura de la pantalla de inicio de sesión.

Interacción con la comunidad

Después de iniciar sesión correctamente, el usuario accede a la pantalla principal de Micelio (ver Figura 19) que incluye un Bottom Tab Navigator o barra inferior de navegación donde el usuario puede acceder a las dos secciones principales de la aplicación, “Comunidad” y “Registros Fungi”.

La primera de ellas y la que se encuentra implementada en su totalidad es “Comunidad”. En esta sección los usuarios pueden interactuar entre sí a través de publicaciones relacionadas al reino Fungi. Las publicaciones existentes se listan y se ordenan de dos formas posibles: por orden de más recientes o por orden de más relevantes.

Los usuarios pueden crear una publicación al presionar el botón “Crear publicación” como se muestra en la Figura 20. Opcionalmente se puede agregar una foto tomada en el momento mediante un componente de cámara personalizado que se ha implementado usando el módulo expo-camera (ver Figura 21).



Figura 19. Captura de la pantalla principal.

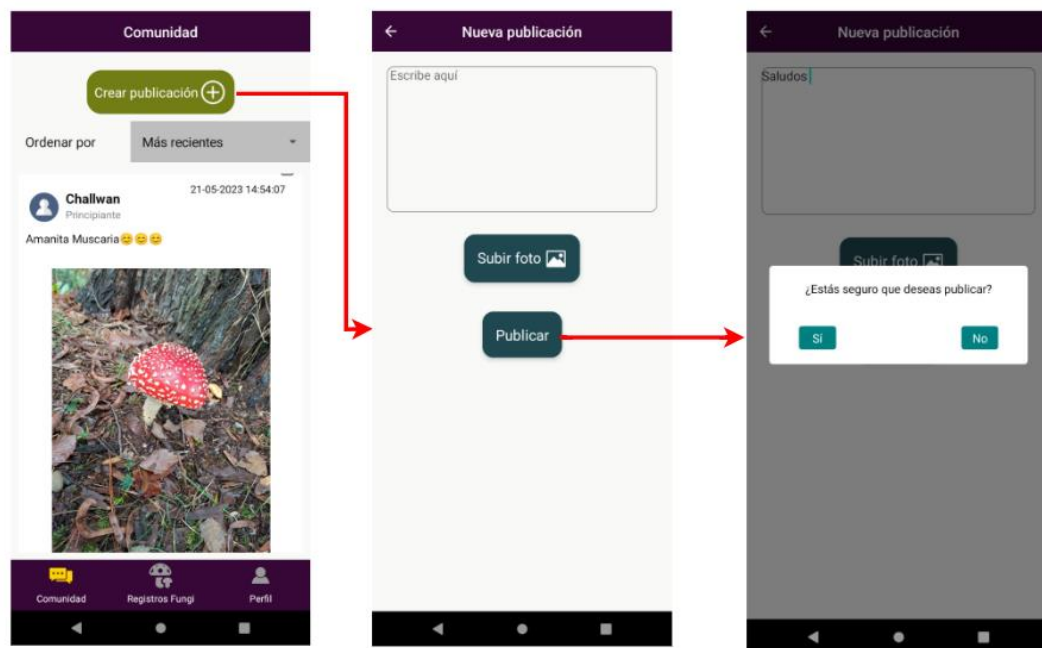


Figura 20. Wireflow para crear una publicación.



Figura 21. Captura de pantalla del componente de cámara.

Los datos de la publicación, incluido el archivo de la imagen son enviados a la API REST usando Axios. A su vez, para desplegar la lista de publicaciones existentes, se hace una solicitud a la API REST donde se obtienen solo los datos básicos de cada publicación. Las imágenes se obtienen por medio de una solicitud independiente que se realiza progresivamente a medida que el usuario desliza la lista de publicaciones. Esto se hace para mejorar el rendimiento de la aplicación y brindar una mejor experiencia de usuario.

Otra funcionalidad implementada en la sección 'Comunidad' de Micelio es la posibilidad de reaccionar y hacer comentarios en las publicaciones, como puede observarse en la Figura 22. Esto permite a los usuarios expresar su opinión, interactuar con otros miembros de la comunidad y fomentar discusiones en torno al reino Fungi.

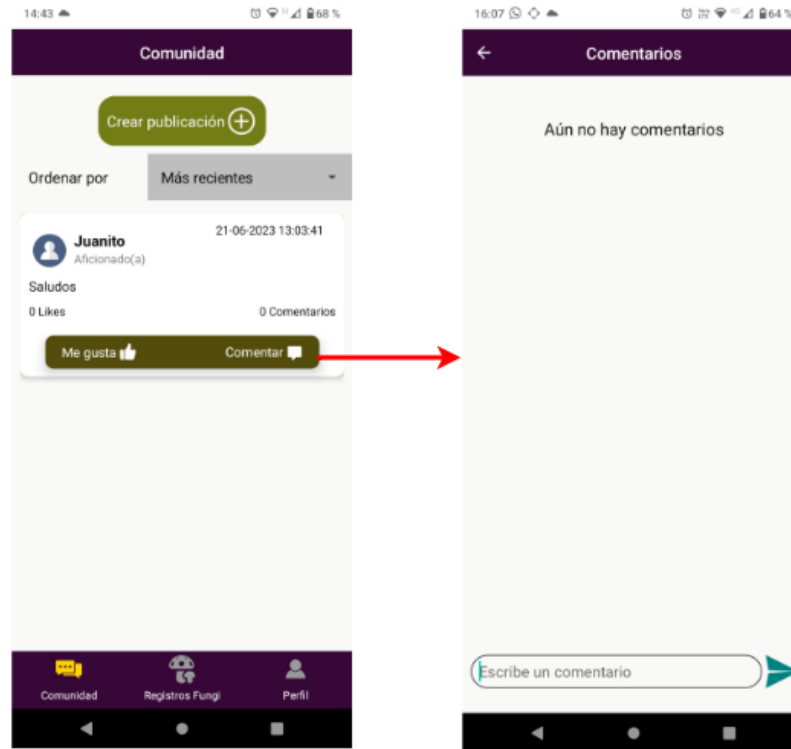


Figura 22. Captura de pantalla de la interfaz de reacciones y comentarios.

Sistema de moderación

Otra funcionalidad implementada durante esta iteración fue el sistema de moderación en Micelio. Para lograr esto, se agregó un nuevo atributo de tipo booleano a cada usuario, el cual indica si posee el rol de moderador. Esta adición permitió distinguir visualmente a los moderadores de los usuarios estándar al renderizar las publicaciones en la plataforma.

Cuando un usuario es designado como moderador, la presentación de las publicaciones se ve ligeramente modificada en comparación con la de los usuarios regulares. En la esquina superior derecha de cada publicación, se generó un botón adicional con un símbolo de "eliminar", como se muestra claramente en la Figura 23. Este botón proporciona a los moderadores una forma rápida y conveniente de tomar medidas disciplinarias en caso de encontrar contenido inapropiado o violatorio de las normas establecidas.

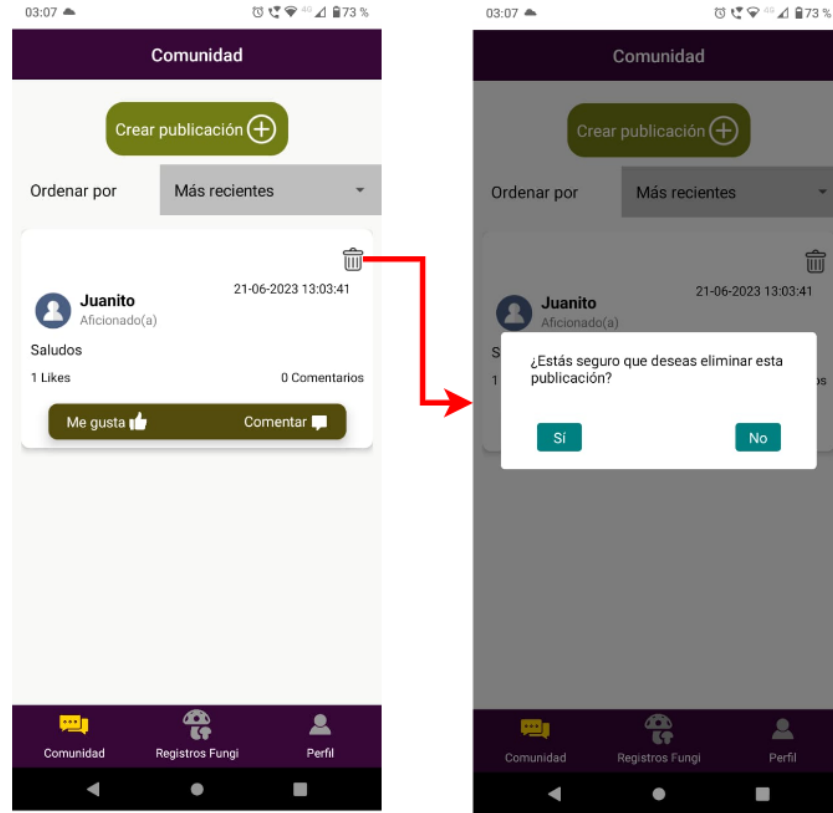


Figura 23. Wireflow de eliminación de una publicación.

Perfil de usuario

Además de todas las características mencionadas anteriormente, el desarrollo de la aplicación incluyó la implementación de una sección de perfil de usuario, que se puede visualizar en la Figura 24. Esta sección tiene como objetivo proporcionar a los usuarios una visión general y personalizada de su actividad en la aplicación, así como acceso rápido a diversas funciones y opciones de configuración.

En este apartado se despliega la información determinada como más relevante para un usuario de la aplicación. Particularmente, se muestra el nombre de usuario, edad, nivel de conocimiento acerca del reino Fungi y la cantidad de publicaciones y registros Fungi realizados. Además, se muestra la opción de cerrar sesión, reportar un problema y ajustes de cuenta.

Adicionalmente, el perfil de usuario incluye opciones importantes para la interacción y gestión de la cuenta. Entre ellas se encuentra la opción de cerrar sesión, que permite a los usuarios finalizar su sesión de forma segura y proteger su privacidad. También se presenta la opción de reportar un problema, brindando a los usuarios una vía directa para informar

cualquier dificultad técnica o incidencia que puedan encontrar en la aplicación y, por último, otras opciones de administración de cuenta.



Figura 24. Captura de pantalla del perfil de un usuario.

5.3 Segundo Sprint: Desarrollo del dispositivo de interacción

5.3.1 Resumen del Sprint

La segunda iteración de desarrollo se enfocó en la construcción del prototipo del dispositivo de interacción compuesto por un microcontrolador Arduino UNO conectado a los sensores de humedad y temperatura. También se trabajó en el frontend de la aplicación móvil relacionado con esta funcionalidad.

5.3.2 Tareas propuestas

En base a los requisitos funcionales definidos previamente, las tareas definidas para el segundo hito de desarrollo se muestran en la Tabla 26.

Tabla 26: Tareas del segundo Sprint.

Tarea	Requisito(s) funcional(es) asociado(s)		Dificultad
Construcción del prototipo	RF8	El sistema debe contar con un microcontrolador que capture y envíe los datos capturados por los sensores	Alta

5.3.3 Resultados del Sprint

Construcción del prototipo

En esta etapa se procede a construir lo será el prototipo del dispositivo de interacción que sea capaz de capturar la temperatura y humedad ambiental, y luego enviar los datos vía Bluetooth y adicionalmente mostrarlos en una pantalla LED. Las herramientas requeridas para comenzar el proceso de construcción y pruebas fueron facilitadas por el profesor copatrocinante. Este set de herramientas incluye una placa Arduino UNO como se observa en la Figura 25, un módulo Bluetooth HC-06 (ver Figura 26), un sensor de humedad y temperatura como se muestra en la Figura 27, una batería externa como fuente de alimentación (ver Figura 28), una pantalla LED tal como se muestra en la Figura 29, una protoboard y elementos básicos para circuitos.

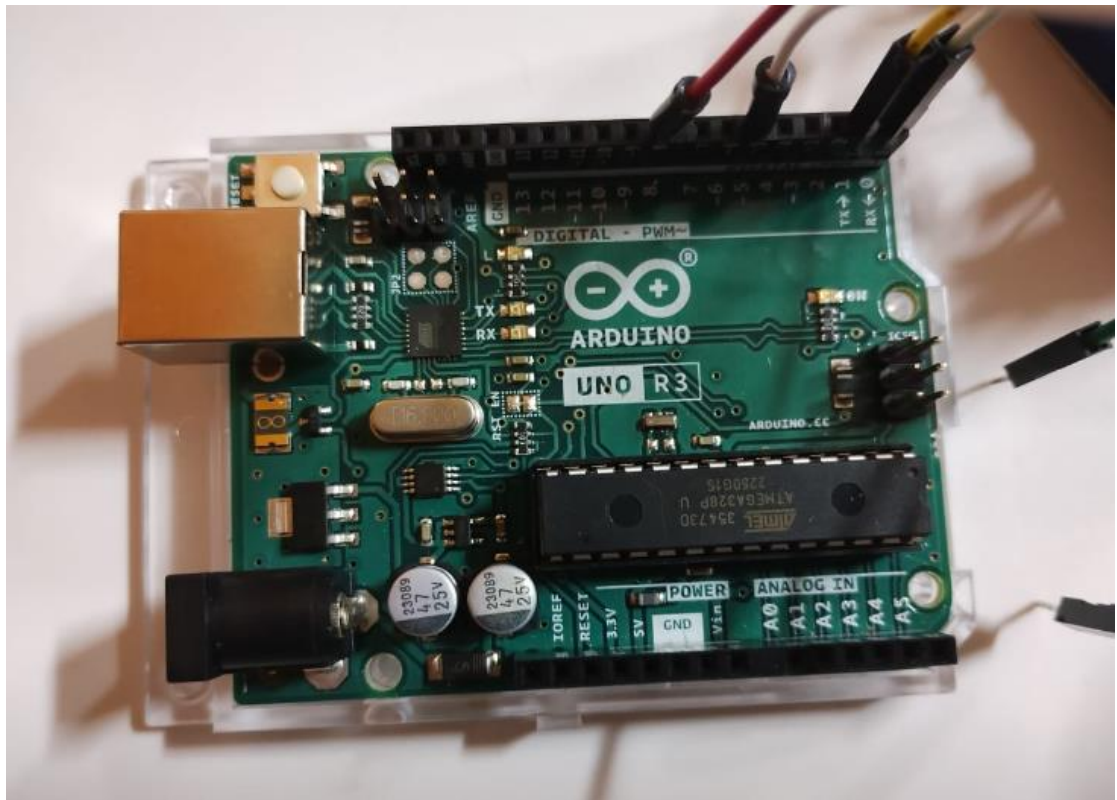


Figura 25. Placa Arduino UNO.



Figura 26. Módulo Bluetooth HC-06.

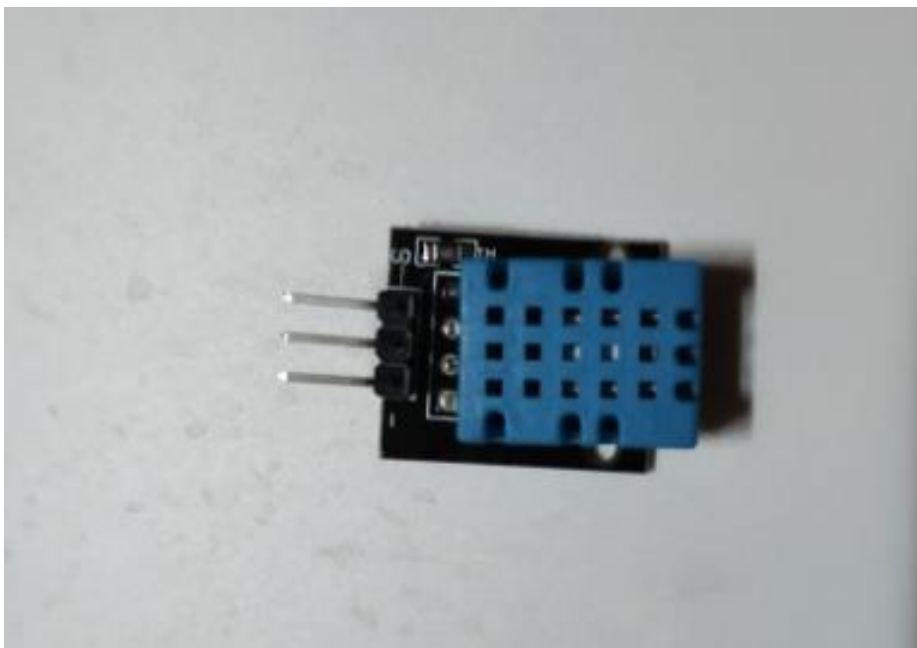


Figura 27. Sensor de humedad y temperatura DHT-11.



Figura 28. Batería externa 5000mAh.



Figura 29. Pantalla LCD I2C 16x2.

Captura de datos de los sensores

A partir de estos componentes, se construyó el primer circuito de prueba para verificar la captura de datos del sensor de humedad y temperatura como se observa en la Figura 30 con un esquema realizado con la herramienta de simulación Circuito.io. Este circuito captura los datos de los sensores y los envía por un canal serial mediante el módulo Bluetooth HC-06. Los esquemas de conexión del sensor DHT-11 y el módulo HC-06 se muestran en la Figura 30 y Figura 31, respectivamente.

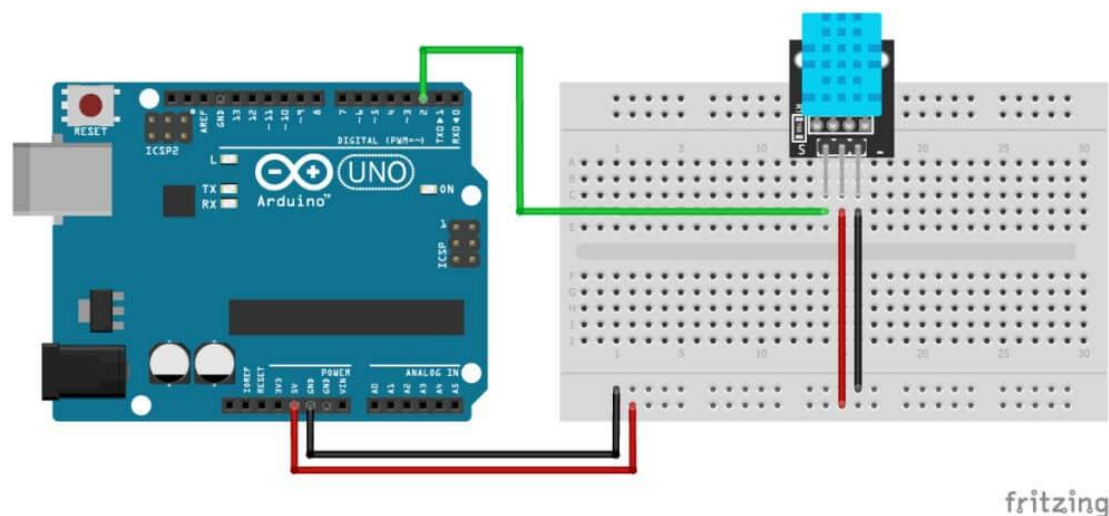


Figura 30. Esquema de conexión del sensor DHT-11.⁵

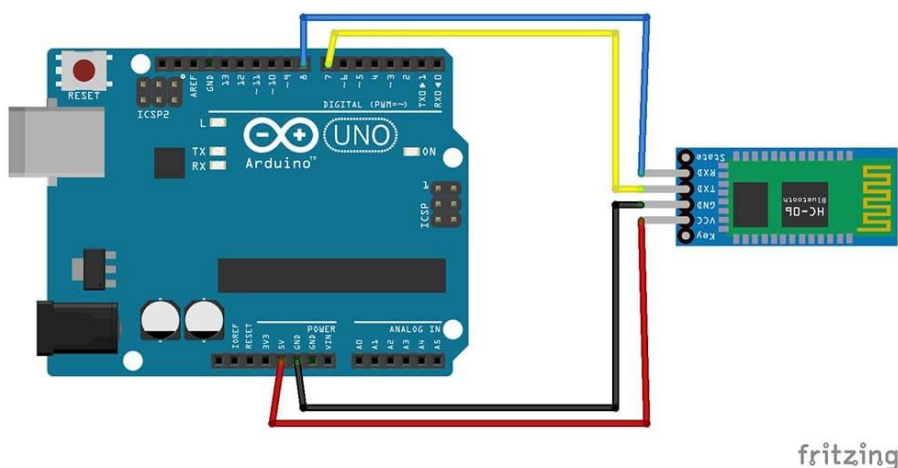


Figura 31. Esquema de conexión del módulo HC-06.⁶

⁵ Benne, 2019. Tutorial del sensor DHT11/DHT12 con Arduino. [Figura]. Recuperado de: <https://www.makerguides.com/es/dht11-dht22-arduino-tutorial/>

⁶ Codizi, 2020. Práctica Bluetooth HC-06. [Figura]. Recuperado de: <https://codiziapp.com/introduccion-a-arduino/practicas/17-practica-bluetooth-hc-06>

El código cargado en la placa Arduino cumple la función definir los parámetros de conexión y captura de datos del sensor, y luego los datos son enviados por un canal serial en intervalos de cinco segundos. A continuación, se muestra el código fuente en cuestión.

```
#include "DHT.h"

#define DHTTYPE DHT11    // DHT 11

const int DHTPin = 5;    // pin de conexion

DHT dht(DHTPin, DHTTYPE);

void setup() {
    Serial.begin(9600);
    dht.begin();
}

void loop() {
    // Retraso de tiempo entre mediciones.
    delay(5000);

    // Lectura de humedad y temperatura
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (isnan(h) || isnan(t)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print(" %\t \n");
    Serial.print("Temperature: ");
    Serial.print(t);
    Serial.print(" *C \n");
}
```

Integrando los esquemas mostrados anteriormente, se procedió a construir este circuito incluyendo el sensor DHT-11 y el módulo HC-06. Posteriormente, el código fuente fue cargado correctamente en la placa Arduino. En la Figura 32 se muestra una foto del circuito funcional. Los resultados de las pruebas se exponen en la sección de Validación y verificación.

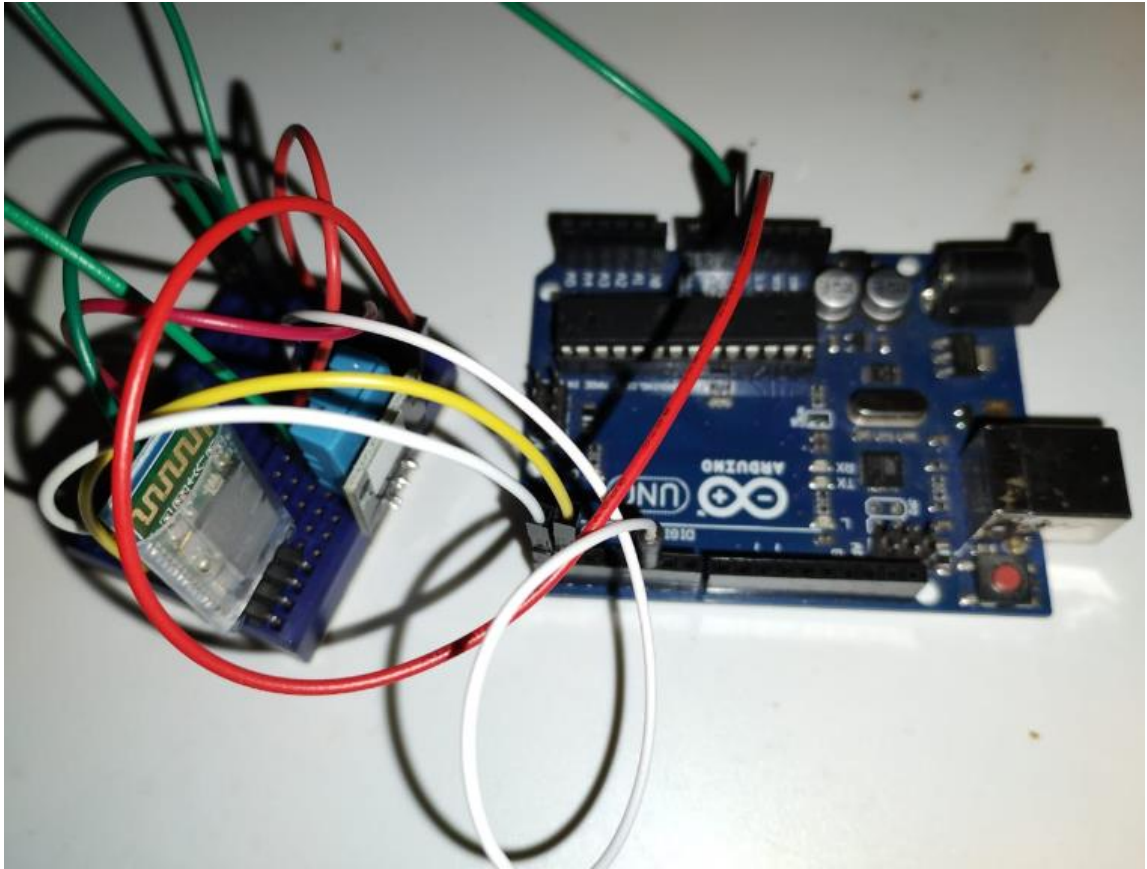


Figura 32. Construcción del primer circuito.

Despliegue de datos en pantalla LCD

El siguiente paso consistió en incorporar la pantalla LCD I2C al circuito principal para establecer una forma complementaria de presentación de los datos capturados por el sensor de humedad y temperatura. El esquema de conexión de la pantalla I2C se muestra en la Figura 33.

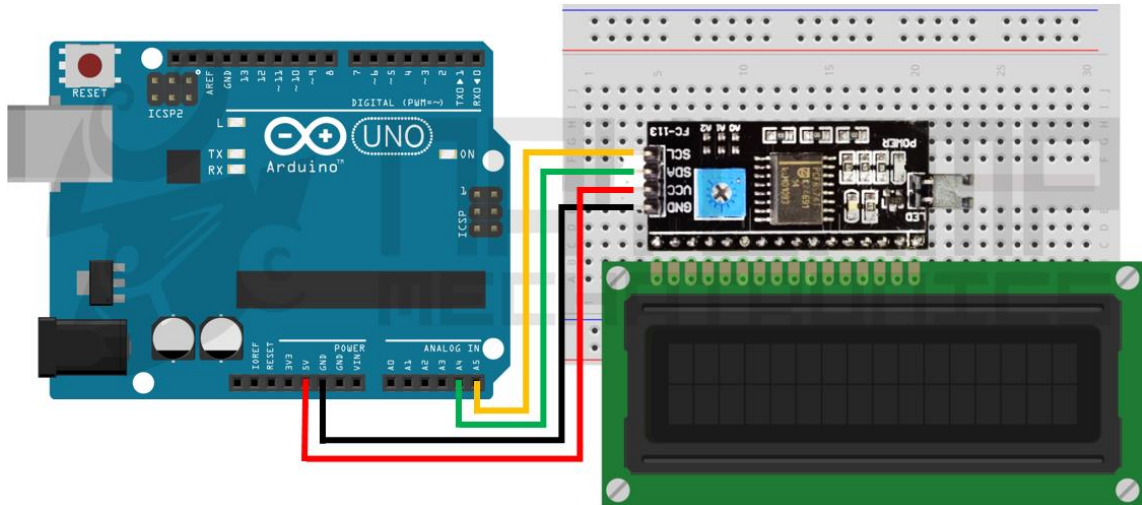


Figura 33. Esquema de conexión de pantalla LCD I2C.⁷

El código fuente cargado en la placa es esencialmente similar al del primer circuito, con la diferencia que en este caso se añadieron las secciones de código destinadas al control de la pantalla LCD. El código fuente completo se muestra a continuación.

```
#include "DHT.h"
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Uncomment whatever type you're using!
#define DHTTYPE DHT11 // DHT 11

LiquidCrystal_I2C lcd(0x27, 16, 2);

const int DHTPin = 5; // what digital pin we're connected to

DHT dht(DHTPin, DHTTYPE);

void setup() {
    lcd.init();
    lcd.backlight();

    Serial.begin(9600);
    Serial.println("DHTxx test!");

    dht.begin();
}
```

⁷ Naylamp Mechatronics, 2023. Tutorial LCD con I2C, controla un LCD con solo dos pines [Figura]. Recuperado de: https://naylampmechatronics.com/blog/35_tutorial-lcd-con-i2c-controla-un-lcd-con-solo-dos-pines.html

```

void loop() {
  // Wait a few seconds between measurements.
  delay(5000);

  // Reading temperature or humidity takes about 250 milliseconds!
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print(" %\t \n");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print(" *C \n");

  lcd.setCursor(0,0);
  lcd.print("Humedad: ");
  //lcd.setCursor(20,0);
  lcd.print(h);

  lcd.setCursor(0,1);
  lcd.print("Temp : ");
  lcd.print(t);
}

```

Posteriormente, el código fuente fue cargado correctamente en la placa Arduino. En la Figura 34 se muestra una foto del circuito funcional. Los resultados de las pruebas de este circuito se exponen en la sección de Validación y verificación.

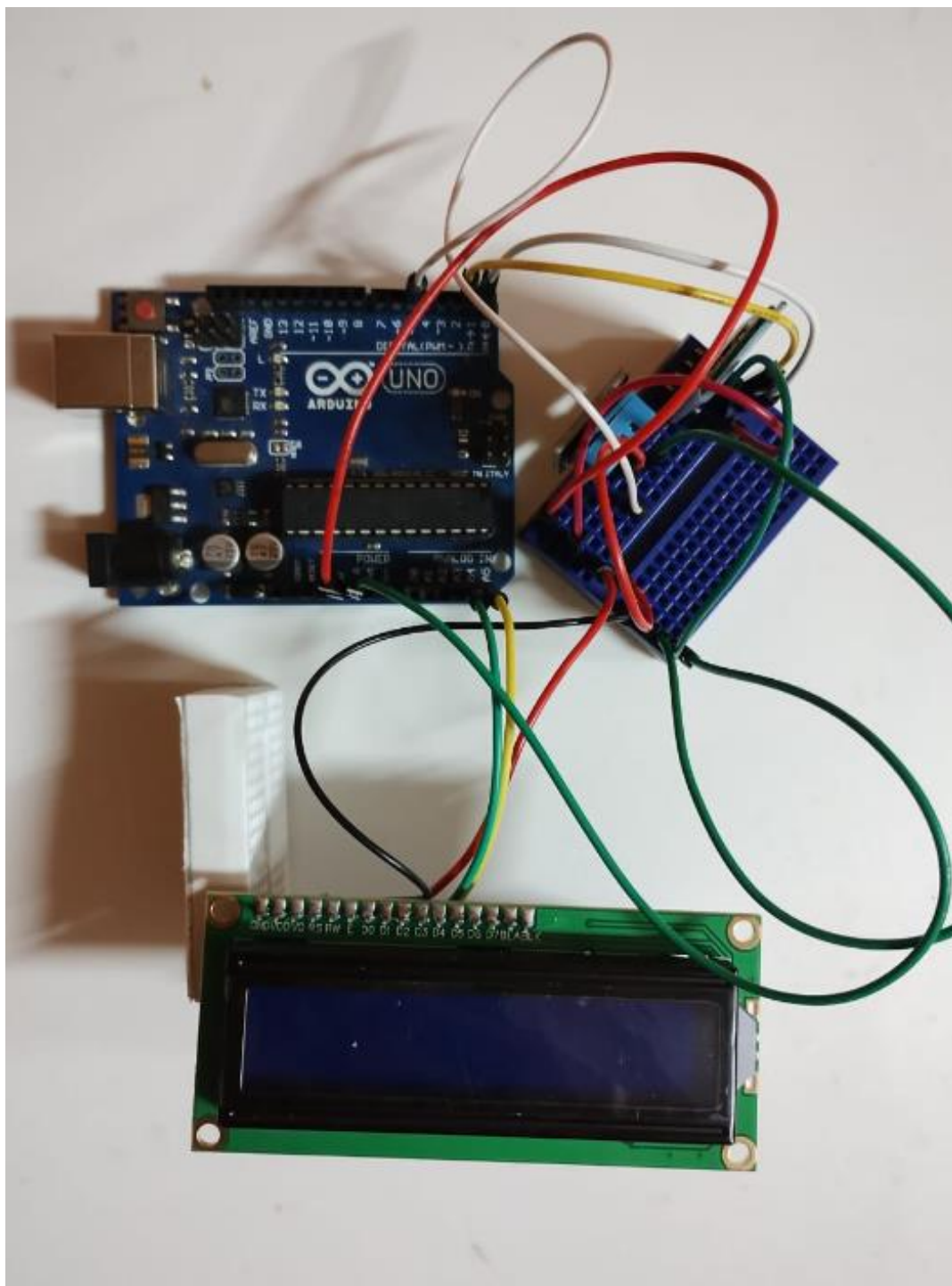


Figura 34. Circuito completo del prototipo de interacción.

5.4 Tercer Sprint: Integración del sistema y mejoras

5.3.1 Resumen del Sprint

Esta última iteración de desarrollo tiene como principal objetivo la integración entre la aplicación móvil y el dispositivo de interacción mediante una interfaz que permita capturar los datos de los sensores y almacenarlos en un registro de la aplicación móvil. Además, en este espacio se implementan mejoras generales a todo el sistema.

5.3.2 Tareas propuestas

De acuerdo con los requisitos funcionales definidos previamente, las tareas definidas para la última iteración de desarrollo se muestran en la Tabla 27.

Tabla 27: Tareas del tercer Sprint.

Tarea	Requisito(s) funcional(es) asociado(s)		Dificultad
Desarrollo frontend relacionado a la interacción con los hongos	RF11	La aplicación debe permitir añadir registros de hongos con todos los datos medidos.	Media
Desarrollo de la captura de datos desde la aplicación	RF9	La aplicación debe permitir la toma de datos mediante sensores de humedad y temperatura.	Alta
Implementación de gráficos de análisis	RF15	La aplicación debe incluir visualizaciones de análisis de datos.	Media

5.3.3 Resultados del Sprint

Desarrollo frontend relacionado a la interacción con los hongos

La segunda sección principal de Micelio y que se desarrolló en este Sprint es “Registros Fungi”. El propósito central de esta sección es lograr una integración efectiva entre la aplicación móvil, el microcontrolador y los sensores de humedad y temperatura, con el objetivo de facilitar una interacción directa con los hongos.

Durante esta iteración, se ha enfocado en el desarrollo de la interfaz de usuario para la sección de "Registros Fungi". La Figura 35 muestra la pantalla principal de esta sección, donde se han aplicado cuidadosamente los principios de diseño para ofrecer una experiencia intuitiva y atractiva a los usuarios.

En esta pantalla principal, los usuarios podrán acceder a dos subsecciones. La primera de ellas consiste en “Agregar registro Fungi”, esta función permite crear un tipo de publicación más especializada de la observación de un hongo. Para ello se muestra un formulario más completo acerca de la observación como se muestra en la Figura 36. Este formulario se divide en tres partes. La primera parte consiste en el ingreso de datos generales como la descripción del registro, localización, y una foto adjunta. La segunda parte permite dar una clasificación visual del tipo de hongo en base a diferentes clases morfológicas más comunes de hongos. Con esto se busca brindar información adicional que pudiese ayudar a identificar el hongo y a la vez, enseñar acerca de las diferentes categorías de hongos. Por último, la tercera parte es aquella donde se capturan los datos medidos con los sensores.

Asimismo, todos los datos del formulario se almacenan en una estructura y, al publicar el registro, se envían a la API REST mediante Axios.



Figura 35. Captura de pantalla inicial de Registros Fungi.

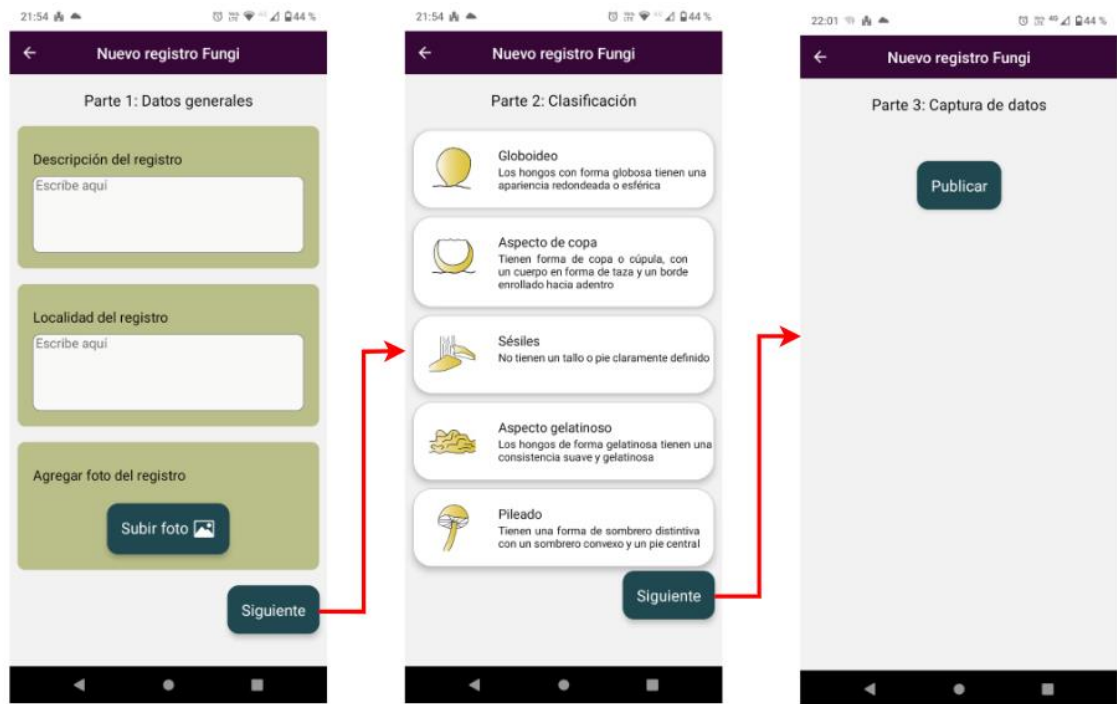


Figura 36. Wireflow de la creación de un registro Fungi.

La segunda subsección “Ver registros” incorpora un concepto similar al de las publicaciones estándar donde todos los registros son listados y los usuarios pueden reaccionar y realizar comentarios que eventualmente podrían ayudar con la clasificación del hongo. En la Figura 37, se muestra un registro Fungi publicado con todos los datos, exceptuando los aquellos de los sensores.



Figura 37. Captura de un registro Fungi publicado.

Desarrollo de la captura de datos desde la aplicación móvil

En primera instancia, se buscaba que la integración del sistema se hiciera mediante Bluetooth. Para ello, se intentó implementar esta funcionalidad de conexión y lectura en la aplicación móvil. Dado que hasta el día de hoy no existe soporte oficial de Expo para el uso de Bluetooth debido a la poca madurez de esta tecnología, fue necesario trabajar con librerías de React Native compatibles con Expo. Para poder utilizar estas librerías fue necesario realizar un proceso denominado “prebuild” donde desde Expo se expone el código nativo de Android permitiendo modificar algunos parámetros específicos y configurar el uso de estas librerías. Para intentar lograr esta integración, se trabajó con las únicas dos librerías disponibles, y que se describen a continuación:

- react-native-ble-plx: Es una librería de código abierto que permite la comunicación y la interacción con dispositivos Bluetooth de baja energía (BLE) en aplicaciones móviles desarrolladas con React Native. Esta librería proporciona una capa de

abstracción sobre las API nativas de Bluetooth de iOS y Android, lo que facilita la implementación de funcionalidades relacionadas con BLE en aplicaciones React Native. Se logró implementar correctamente la librería en el proyecto, sin embargo, hubo un problema de comunicación con el módulo Bluetooth HC-06 dado que este utiliza Bluetooth Classic, el cual es un protocolo distinto a BLE en términos de comunicación. Esto ocasionó que el módulo HC-06 no pueda ser detectado por el escaneo de dispositivos Bluetooth de react-native-ble-plx, por lo tanto, no pudo establecerse conexión.

- react-native-bluetooth-classic: La segunda opción explorada fue react-native-bluetooth-classic que al igual que la alternativa anterior es una librería de React Native que proporciona una interfaz para interactuar con dispositivos Bluetooth Classic en aplicaciones móviles. Al implementar esta librería en el proyecto se logró detectar el dispositivo HC-06 mediante un escaneo Bluetooth, sin embargo, al intentar establecer conexión se presentaron ciertas limitaciones por parte de react-native-bluetooth-classic ya que no era posible vincularse e iniciar una comunicación serial con el módulo HC-06.

Dado que no fue posible lograr esta integración mediante Bluetooth, se optó por incluir la funcionalidad de que el usuario pueda ingresar manualmente los datos capturados del sensor y que son desplegados en la pantalla LCD tal como se observa en la Figura 38.

The screenshot shows a mobile application interface with a dark purple header bar containing a back arrow and the text 'Nuevo registro Fungi'. Below the header, the text 'Parte 3: Captura de datos' is centered. Underneath, the text 'Medición de la humedad y temperatura de los sensores' is displayed. There are two input fields: 'Temperatura(°C):' with a placeholder 'Escribe aquí' and 'Humedad(%):' with a placeholder 'Escribe aquí'. A dark green button labeled 'Publicar' is positioned below the input fields. The bottom of the screen shows the standard Android navigation bar.

Figura 38. Captura de pantalla del ingreso de temperatura y humedad.

Con esto, se completa la integración de todo el sistema y la creación de un registro Fungi desde la aplicación móvil con los datos descriptivos, de localización, de clasificación, y los datos capturados de los sensores se vuelve completamente funcional. En la Figura 39, se muestra un diagrama del proceso de creación de un registro Fungi creado con el software Bizagi y que incluye toda la sucesión de tareas necesarias para la completitud de esta actividad.

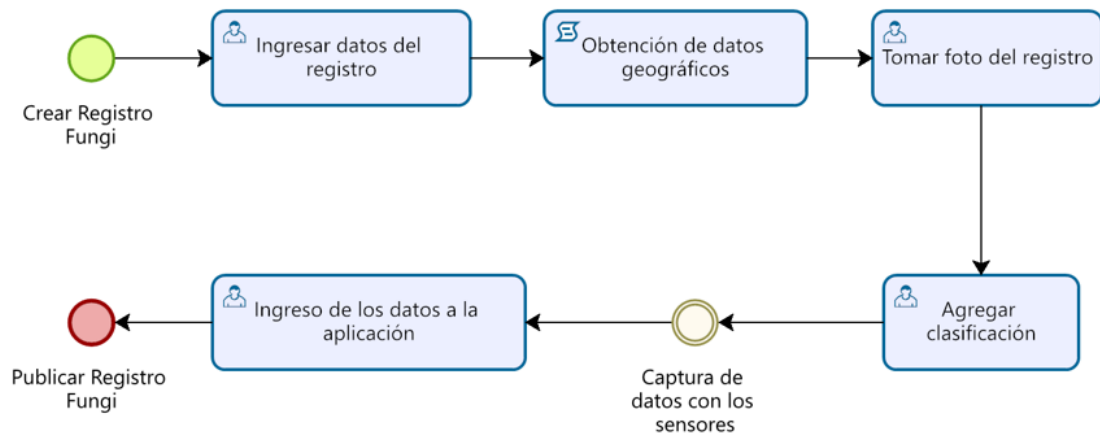


Figura 39. Diagrama de proceso BPMN de la creación de un registro Fungi.

Implementación de gráficos de análisis

En el ámbito del análisis de datos, se optó por incluir una nueva subsección denominada “Analítica” accesible desde la sección “Registros Fungi”. La finalidad es que los usuarios puedan obtener información adicional y aprender a partir de los datos existentes en cuanto a registros Fungi. Esta implementación se logró con la librería react-native-chart-kit, la cual es bastante completa en cuanto a tipos de gráficos y personalización. Se desarrollaron dos gráficos distintos como se observa en la Figura 40, de momento poblados con datos de prueba ya que no se cuentan con suficientes registros en la base de datos.

El primer gráfico es de tipo lineal y permite visualizar la cantidad de registros existentes por cada mes del año, de tal forma que se puedan detectar algunos patrones o tendencias. Puede ser una herramienta útil para detectar fluctuaciones estacionales o períodos de mayor actividad en la recolección de datos.

El segundo gráfico es de tipo circular y muestra el porcentaje de cada una de las clases de hongos definidas en la aplicación. Se muestra el porcentaje de cada clase en relación con el total de registros. Así se podrá tener una comprensión clara de la distribución de las clases de hongos y su importancia relativa en la base de datos. Esta visualización ayudará a identificar las clases más comunes o destacadas en los registros.

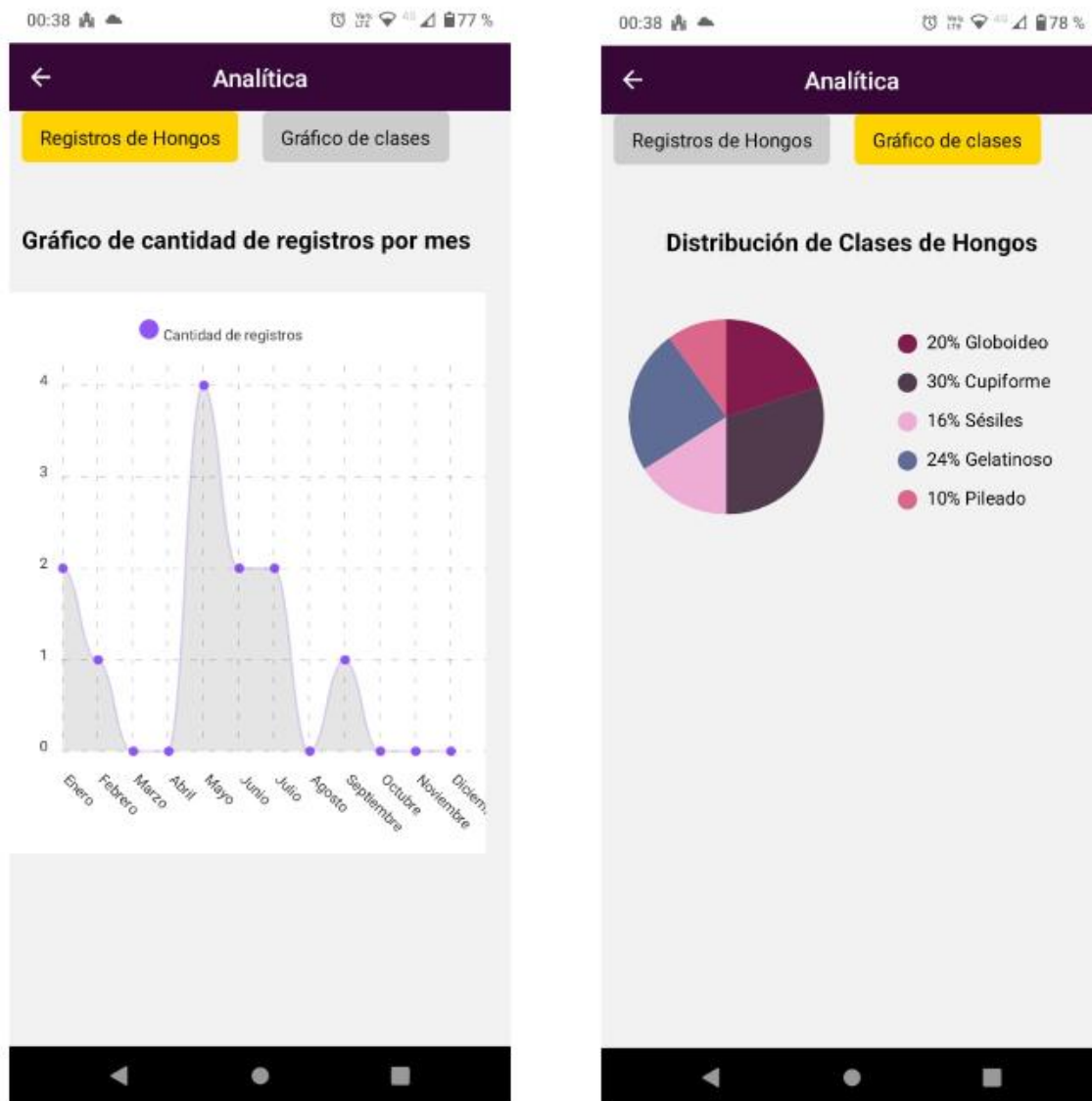


Figura 40. Gráficos implementados con datos de prueba.

5.5 Validación y verificación

En este apartado, se presentan las pruebas de validación del sistema que se llevaron a cabo para asegurar la calidad y el correcto funcionamiento del software desarrollado.

5.5.1 Prueba N°1: Envío de datos vía Bluetooth

Esta prueba tiene por objetivo verificar la transmisión de datos desde Arduino por el canal serial. Para testear esta funcionalidad, se usó una aplicación móvil genérica desarrollada

con App Inventor (Villalpando, 2015). Esta aplicación permite vincularse a un dispositivo Bluetooth HC-06 y recibir los datos enviados mediante este canal. Para esta prueba se utilizó el circuito con el sensor DHT-11 para realizar las mediciones de humedad y temperatura, y el módulo HC-06 para enviar los datos vía Bluetooth. El primer paso consistió en la construcción del circuito como se observa en la Figura 41.

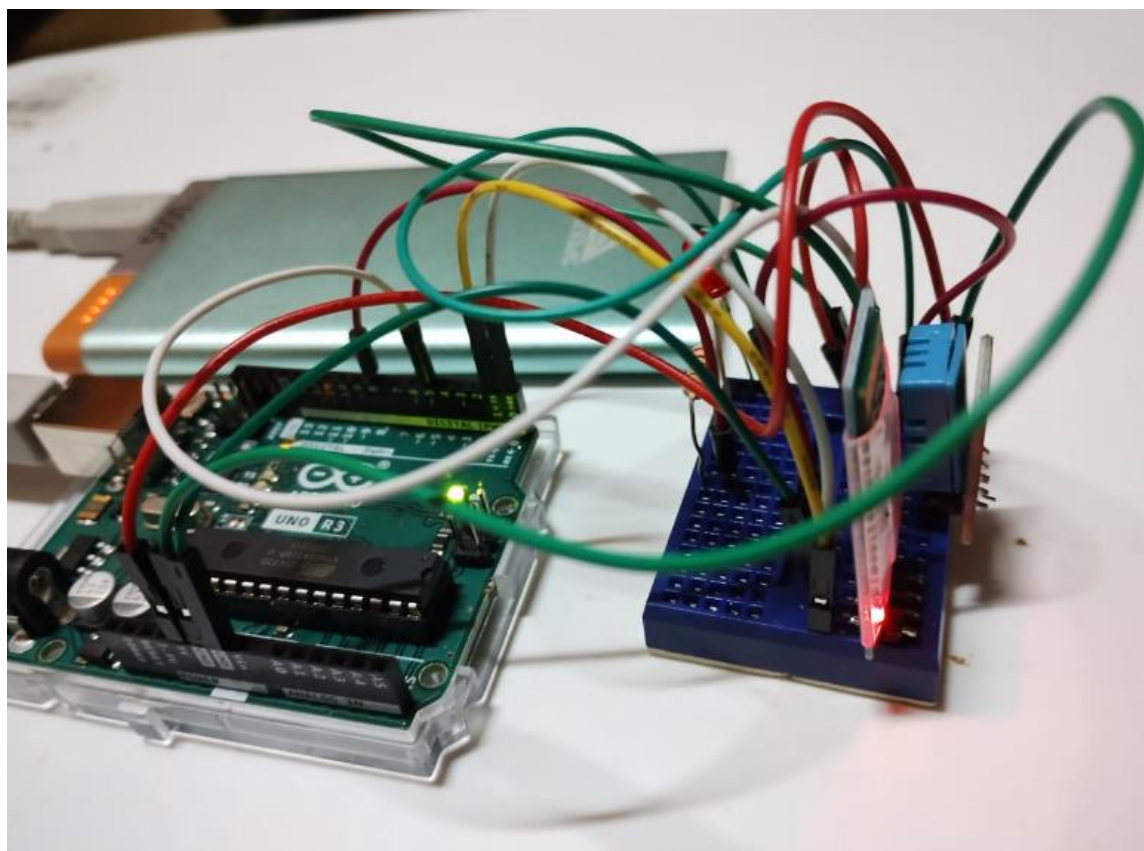


Figura 41. Circuito Prueba N°1.

Luego, se enciende el Bluetooth del teléfono y se inicia la aplicación, la cual cuenta con una interfaz muy simple como se observa en la Figura 42.

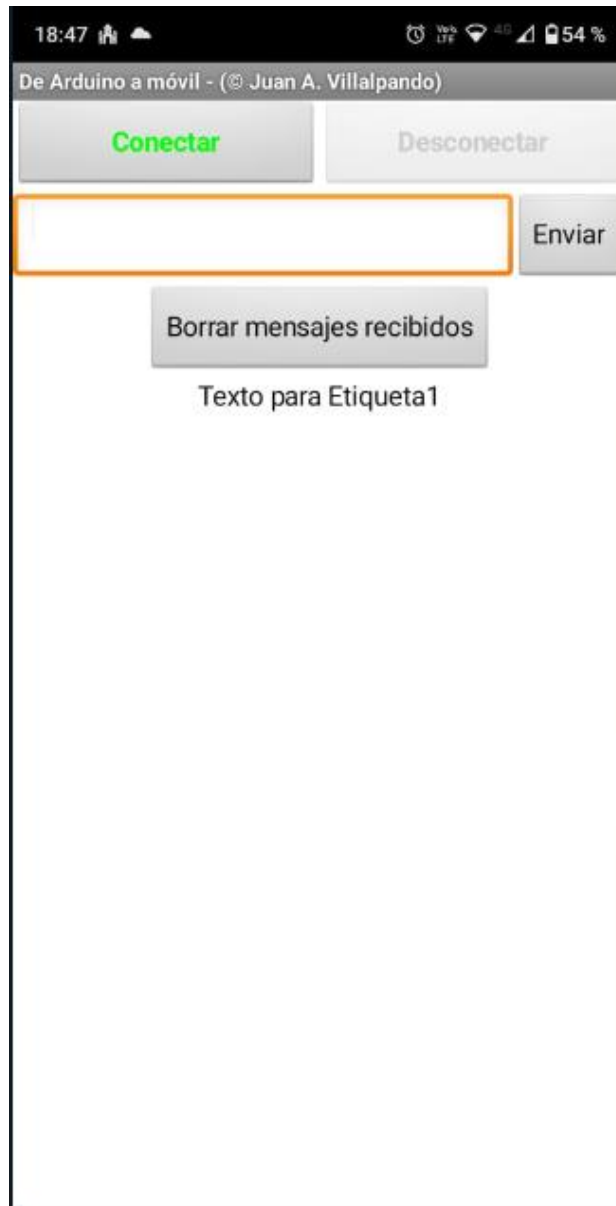


Figura 42. Captura de la aplicación móvil utilizada en la prueba.

Posteriormente, se efectúa la conexión con el dispositivo HC-06 luego de presionar el botón “Conectar” y presionar el dispositivo correspondiente como se observa en la Figura 43.

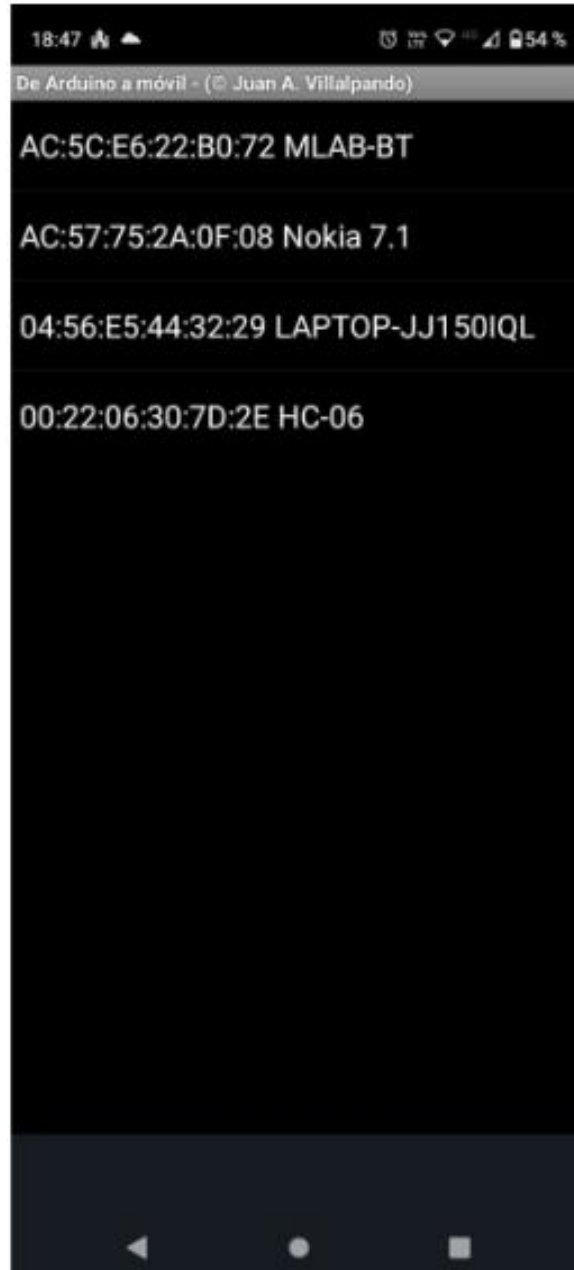


Figura 43. Captura de pantalla de la conexión al dispositivo HC-06.

Finalmente, se establece correctamente la conexión y se comienzan a recibir los datos del sensor DHT-11 en intervalos de cinco segundos tal como se observa en la Figura 44.

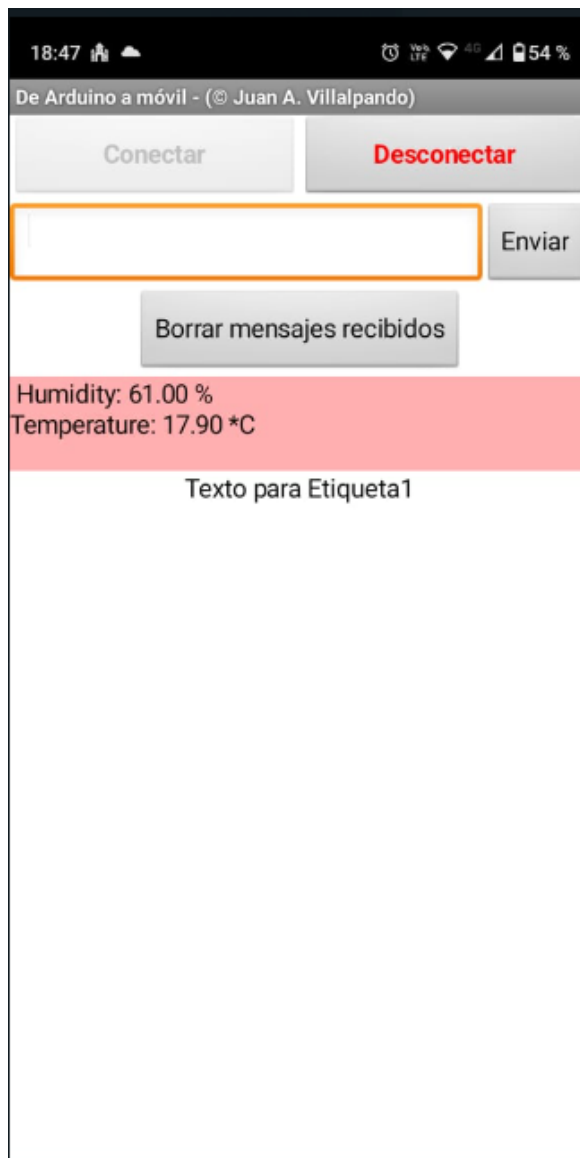


Figura 44. Captura de pantalla de los datos medidos por el sensor.

En base a estos resultados, se verifica que los datos están siendo enviados correctamente y, por lo tanto, se puede concluir que se ha pasado esta prueba con éxito.

5.5.2 Prueba N°2: Despliegue de datos en la pantalla LCD

Esta prueba tiene por objetivo verificar que los datos capturados por el sensor DHT-11 se muestren correctamente en la pantalla. Para ello, se añade la pantalla LCD I2C al circuito con el sensor como se muestra en la Figura 45.

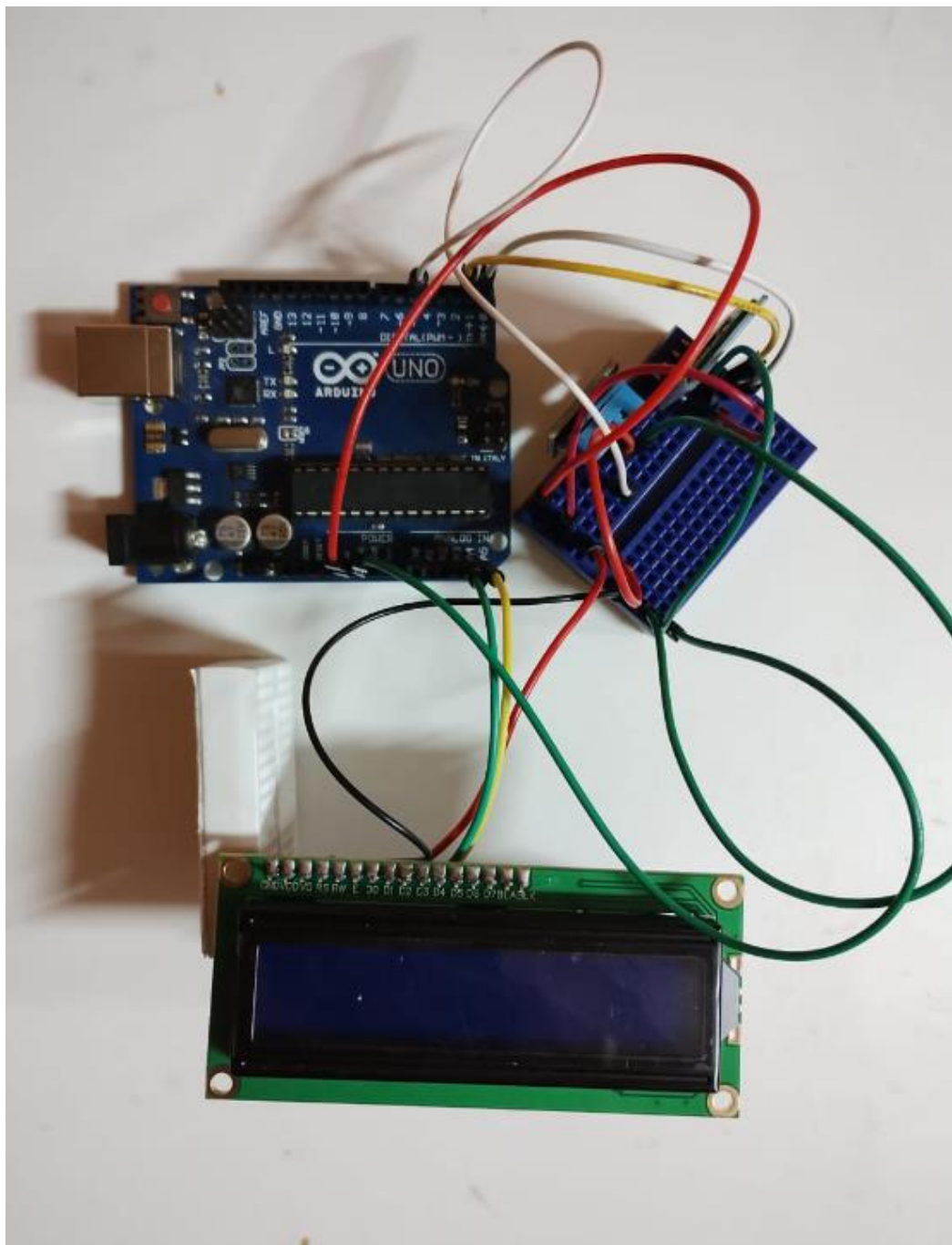


Figura 45. Circuito Prueba N°2.

Luego, en la Figura 46, se pueden observar los datos mostrados en la pantalla LCD.



Figura 46. Despliegue de datos en la pantalla LCD I2C.

Por lo tanto, se verifica que los datos obtenidos del sensor se muestran correctamente en la pantalla LCD, por lo tanto, se puede concluir que se ha pasado esta prueba con éxito.

5.5.3 Prueba N°3: Funcionamiento del sistema

La última prueba consiste en poner a prueba el funcionamiento de todo el sistema y su integración. Para ello, se simula la creación de un registro Fungi en un entorno real siguiendo todo el flujo de tareas que se requiere para ello, partiendo desde la aplicación móvil e incorporando el prototipo de interacción para capturar los datos del sensor.

El primer paso consistió en acceder a Micelio y navegar hasta la sección “Registros Fungi” como se observa en la Figura 47, para posteriormente presionar el botón “Agregar registro Fungi” y comenzar el proceso.



Figura 47. Captura de pantalla de Registros Fungi.

El siguiente paso consiste en comenzar a completar el formulario del registro. Para ello se definieron los siguientes datos de prueba mostrados en la Tabla 28.

Tabla 28. Datos de prueba para el registro.

Descripción	Registro Fungi Test
Ubicación	Valdivia
Clasificación	Pileado

En la Figura 48 se muestra el ingreso de los datos de prueba en el formulario de creación del registro Fungi. Adicionalmente, se toma una foto genérica de un hongo ya que es un requisito para poder subir el registro.

The screenshot shows a mobile application interface for creating a new fungal record. At the top, a dark purple header bar contains a back arrow and the title 'Nuevo registro Fungi'. Below this, the section 'Parte 1: Datos generales' is displayed. The form consists of three main input areas, each with a light green header and a white text box. The first area is for 'Descripción del registro' with the text 'Registro Fungi Test'. The second area is for 'Localidad del registro' with the text 'Valdivia'. The third area is for 'Agregar foto del registro', which shows a confirmation message 'Foto cargada correctamente' with a green checkmark and a 'Subir foto' button with a camera icon. At the bottom right of the form is a 'Siguiete' button. The status bar at the very top shows the time as 03:24 and a battery level of 71%.

Figura 48. Captura de pantalla del formulario con los datos de prueba.

Luego, se da clasificación visual al registro y tal como se definió en los datos de prueba, se selecciona “Pileado”.



Figura 49. Captura de pantalla de la parte de clasificación.

A partir de este punto es necesario incorporar el dispositivo de interacción para capturar los datos de humedad y temperatura con el sensor. Para ello, se instaló el circuito conectado a su fuente de alimentación como se observa en la Figura 50. En la Figura 51, se muestran los datos medidos en el momento de la prueba.

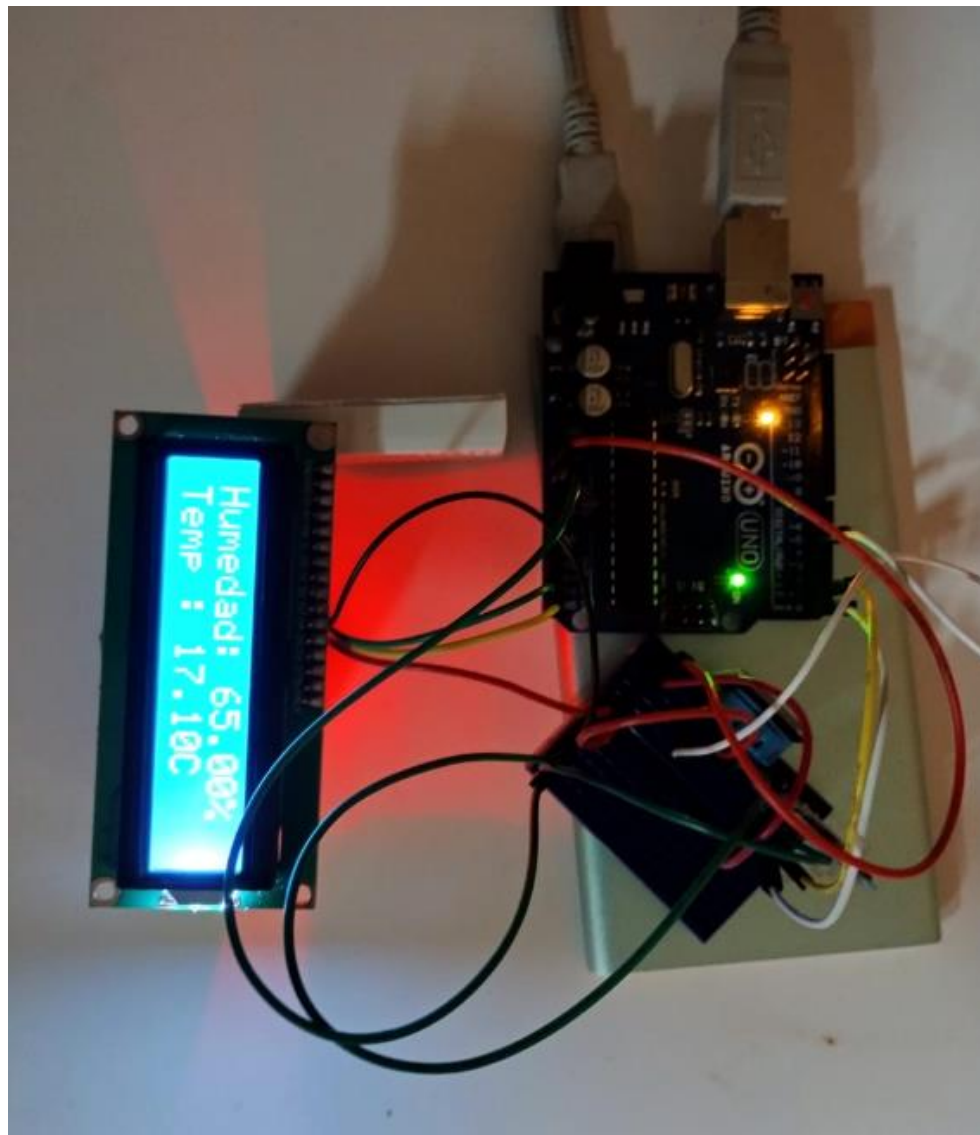


Figura 50. Instalación del prototipo.



Figura 51. Datos medidos por el sensor.

Una vez medidos los datos de temperatura, se vuelve a la aplicación móvil y se ingresan manualmente en la sección de captura de datos como se muestra en la Figura 52.

The screenshot displays a mobile application interface titled "Nuevo registro Fungi". At the top, there is a status bar showing the time as 03:38 and battery level at 69%. Below the title bar, the section is labeled "Parte 3: Captura de datos". The main heading within this section is "Medición de la humedad y temperatura de los sensores". There are two input fields: "Temperatura(°C):" with the value "17.1" and "Humedad(%):" with the value "65.0". A green "Publicar" button is positioned below the input fields. At the bottom of the screen, a numeric keypad is visible, featuring buttons for digits 1-9, 0, a decimal point, a backspace key, and a confirmation key with a checkmark.

Figura 52. Captura de pantalla del ingreso de los datos del sensor.

Tras haber ingresado todos los datos requeridos, se procedió a publicar en registro. Una vez que la aplicación indicó que el registro se subió correctamente, se verificó en la sección "Ver registros" como se muestra en la Figura 53.



Figura 53. Captura de pantalla del registro de prueba publicado.

Como se observa en la Figura 53, el registro contiene todos los datos ingresados, adicionando los datos específicos de ubicación que se obtienen pasivamente. En esta sección, los registros desplegados son recuperados directamente desde la API REST, lo que indica que la comunicación con el servidor y el manejo de peticiones funcionan correctamente. En base a esto, se puede verificar que el sistema es funcional y se puede concluir que se ha superado la prueba.

6. CONCLUSIONES Y TRABAJOS FUTUROS

Conclusiones

Luego de exponer los resultados obtenidos a lo largo de este documento, se puede decir que con el desarrollo de este trabajo se logró completar satisfactoriamente el objetivo principal, el cual consistía en “Desarrollar una aplicación móvil que incorpore un modelo de conformación de red en integre lo físico y lo digital para fortalecer la relación Humano-Hongo y divulgar la importancia del reino Fungi a la comunidad”, ya que si bien el sistema se encuentra a nivel de prototipo, es una propuesta que logra vincular a las personas con el reino Fungi de una forma innovadora, además se destaca que todo el desarrollo fue asistido y retroalimentado por profesionales de ambas áreas involucradas como son la informática y la micología. Respecto a los objetivos específicos, de igual forma se concluye que se han completado según lo propuesto. Por un lado, se tiene el modelo de red social que se ha implementado en la aplicación móvil que permite las principales formas de interacción entre usuarios de esta red. Además, se ha logrado integrar lo físico y lo digital teniendo como nexo el dispositivo de interacción construido. Por último, durante el desarrollo de la aplicación, se han seguido buenas prácticas de diseño y usabilidad para lograr una experiencia de usuario satisfactoria.

En cuanto al desarrollo, la implementación de arquitecturas limpias siguiendo el enfoque de Domain Driven Design resultó ser una gran ventaja en términos de organización y legibilidad del código. Al aplicar estas prácticas, pude establecer una estructura sólida y coherente en el proyecto, lo cual facilitó la comprensión y mantenimiento del código a lo largo del tiempo.

Considero que uno de los mayores desafíos enfrentados durante el desarrollo tiene relación con la madurez de algunas de las tecnologías utilizadas como es el caso de Expo, ya que como se mencionó anteriormente es un framework relativamente nuevo y aún no cuenta con soporte oficial para diversas funcionalidades. Por ello, no fue posible establecer la integración por Bluetooth entre la aplicación y el dispositivo de interacción. Esto pudo haberse evitado en fases tempranas del proyecto seleccionando alguna tecnología de desarrollo nativa y con un mayor grado de madurez como por ejemplo Android Studio.

En general, el desarrollo de este trabajo fue una experiencia valiosa en diversos sentidos, comenzando por el hecho de ser un proyecto interdisciplinario donde se trabajó con personas de diferentes áreas ajenas a la informática, lo que favoreció la lluvia de ideas y la discusión. Además, este proyecto de software me brindó una valiosa oportunidad para adentrarme en áreas de desarrollo que no había explorado completamente a lo largo de mi carrera. Específicamente, tuve la posibilidad de incursionar en el desarrollo móvil y el desarrollo backend, lo cual me permitió adquirir conocimientos y experiencia en estas disciplinas. Esta experiencia amplió mi conjunto de habilidades y me proporcionó valiosas lecciones que podré aplicar en futuros proyectos.

Trabajos futuros

Con respecto a trabajos futuros relacionados a este proyecto, se considera la opción de construir un prototipo del dispositivo de interacción totalmente empaquetado y replicable con ayuda de impresión 3D, que puedan ser entregados a diferentes parques y jardines botánicos para que puedan ser utilizados por los visitantes en conjunto con la aplicación.

La aplicación móvil Micelio tiene gran potencial de crecimiento y mejoras, en base a esto se espera que pueda ser tomada por otro tesista para implementar nuevas funcionalidades de interacción, de analítica, soporte multiplataforma, etc.

7. BIBLIOGRAFÍA

- Arduino Documentation. Accedido el 10 de junio, 2023, desde <https://docs.arduino.cc/learn/starting-guide/whats-arduino>
- Barrientos-Avendaño, E., Rico-Bautista, D., Coronel-Rojas, L. A., & Cuesta-Quintero, F. R. (2019). Jardín botánico: Prototipo de software para la gestión y divulgación de plantas nativas basado en código QR y realidad aumentada. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (E17), 267-282.
- Chirica, D. B., & Iftene, A. (2021). Enhancing the Visit of the Botanical Garden of Iasi with the Help of Smart Devices and Augmented Reality. In *Proceedings of the Conference on Mathematical Foundations of Informatics MFOI2020*, Kyiv, Ukraine (pp. 78-87).
- Coelho, A., van Zeller, M., Cardoso, P., Santos, L., Vaz, R., & Raimundo, J. (2020). Gamifying the Museological Experience. In *XCR* (pp. 5-8).
- Expo: FAQ. Accedido el 10 de mayo, 2023, desde <https://docs.expo.dev/faq/>.
- Gimeno, A. (2002). Principales factores condicionantes para el desarrollo de los hongos y la producción de micotoxinas. *Engormix*. Accedido el 16 de abril, 2023, desde <https://www.engormix.com/micotoxinas/articulos/principales-factores-condicionantes-desarrollo-t26065.htm>.
- GlobalStats' Statcounter. Mobile Operating System Market Share Worldwide. Accedido el 10 de mayo, 2023, desde <https://gs.statcounter.com/os-market-share/mobile/worldwide>.
- Guillen Lozano, F. A., & Zapata Galarza, K. S. (2022). Aplicación móvil para el aprendizaje del reino fungi en bosques de selva alta con realidad aumentada, gamificación y microlearning.
- IBM. What is Docker?. Accedido el 30 de mayo, 2023, desde <https://www.ibm.com/topics/docker>.
- Ibrahim, N. Z., Osman, R., Ali, A. M., Hamid, N. H. A., Nordin, S., Shahimi, S., ... & Razak, F. A. (2022). Integrating Augmented Reality in Learning Plants. In *IOP Conference Series: Earth and Environmental Science* (Vol. 1019, No. 1, p. 012051). IOP Publishing.
- Islam, M. A., Islam, M. A., Miah, M. S. U., & Bhowmik, A. (2022). An automated monitoring and environmental control system for laboratory-scale cultivation of oyster mushrooms using the Internet of Agricultural Thing (IoAT).

- In Proceedings of the 2nd International Conference on Computing Advancements (pp. 207-212).
- ITMadrid. (2020). Qué es y para qué sirve Design Thinking. Accedido el 22 de abril de 2023 desde: <https://www.itmadrid.com/que-es-y-para-que-sirve-design-thinking/>.
- Express web framework (Node.js/JavaScript). MDN web docs. Accedido el 10 de mayo, 2023, desde https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs.
- Liu, J., Byrne, D., & Devendorf, L. (2018). Design for collaborative survival: An inquiry into human-fungi relationships. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (pp. 1-13).
- MariaDB Foundation. Accedido el 11 de mayo, 2023, desde <https://mariadb.org>.
- Mauricio Yagui, M. M., Monsore Passos Maia, L. F., Oliveira, J., & Vivacqua, A. (2019). A Crowdsourcing Platform for Curating Cultural and Empirical Knowledge. A Study Applied to Botanical Collections. In IEEE/WIC/ACM International Conference on Web Intelligence-Companion Volume (pp. 322-326).
- Nobnop, R., Wongwatkit, C., & Soponronnarit, K. (2018). A development of 3D augmented reality mobile application to facilitating ecotourism-based herbal learning in MFU botanical garden.
- Postolache, S., Torres, R., Afonso, A. P., Carmo, M. B., Cláudio, A. P., Domingos, D., ... & Redweik, P. (2022). Contributions to the design of mobile applications for visitors of Botanical Gardens. *Procedia Computer Science*, 196, 389-399.
- Rothschuh, U. (2021). Estructura de los hongos. *Ecología Verde*. Accedido el 16 de abril, 2023, desde <https://www.ecologiaverde.com/estructura-de-los-hongos-3676.html>.
- Schwaber, K., & Sutherland, J. (2020). The Scrum Guide. Accedido el 5 de mayo, 2023 desde <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100>.
- Scott, C. (2022). What is Android? Here's everything you need to know. Accedido el 10 de mayo, 2023, desde <https://www.androidauthority.com/what-is-android-328076/>.
- Suconel. (2023). Factores condicionantes para el desarrollo de los hongos. Accedido el 17 de abril, 2023, desde <https://suconel.com/factores-condicionantes-para-el-desarrollo-de-los-hongos/>.
- Ulloa, R. (2019). Desarrollo de una plataforma web y aplicación móvil para el control

automatizado de un invernadero. Universidad Austral de Chile.

- Villalpando, J. (2015). Cómo programar los teléfonos móviles con Android. Bluetooth, Arduino y App inventor 2. Accedido el 8 de junio, 2023, desde <http://kio4.com/appinventor/9bluetootharduino.htm>
- Winter, A., Pedro, E., Ślasko, J., Battaglini, J., Faelker, M., Kivipelto, R., ... & Guedes, P. (2019). Waste to Fungi: An EPS@ ISEP 2019 Project. In Proceedings of the Seventh International Conference on Technological Ecosystems for Enhancing Multiculturality (pp. 115-122).