**[S.I.N.H.2.O]**

**(SAD) Software Architecture Document**

**Versión 2.0**

**Identificación de Documento**

|  |  |
| --- | --- |
| **Identificación** | Nombre del equipo |
| **Proyecto** | Sistema de irrigación por nivel de humedad de 2 orientaciones (S.I.N.H.2.O) |
| **Versión** | 2.0 |

|  |  |
| --- | --- |
| **Documento mantenido por** | Matías Mora |
| **Fecha de última revisión** | 11/09/2024 |
| **Fecha de próxima revisión** |  |

|  |  |
| --- | --- |
| **Documento aprobado por** |  |
| **Fecha de última aprobación** |  |

**Historia de Revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 11/09/2024 | 1.0 | Introducción, Contexto del Problema y Propósito, Ámbito, Definiciones, acrónimos y abreviaciones, referencias, representación, Metas de la arquitectura, Restricciones de la arquitectura, otros antecedentes y consideraciones. | Matías Mora, Sebastian Pino. |
| 12/09/2024 | 2.0 | Especificación de Casos de Uso Relevantes, Especificación de los Escenarios de Calidad Relevantes | Matías Mora, Sebastian Pino y Andrés Silva. |
|  |  |  |  |
|  |  |  |  |

**Tabla de Contenidos**

Contenido

[1 Introducción 3](#_Toc179034651)

[1.1 Contexto del Problema y Propósito 3](#_Toc179034652)

[1.2 Ámbito 3](#_Toc179034653)

[1.3 Definiciones, acrónimos y abreviaciones 3](#_Toc179034654)

[1.4 Referencias 4](#_Toc179034655)

[1.5 Representación 4](#_Toc179034656)

[2 Metas y Restricciones de la Arquitectura 4](#_Toc179034657)

[2.1 Metas de la arquitectura 4](#_Toc179034658)

[3 Restricciones de la Arquitectura 4](#_Toc179034659)

[3.1 Otros antecedentes y consideraciones 5](#_Toc179034660)

[4 Vista de Casos de Uso y Escenarios de Calidad 5](#_Toc179034661)

[4.1 Modelo de Casos de Uso 5](#_Toc179034662)

[4.2 Especificación de Casos de Uso Relevantes 5](#_Toc179034663)

[4.3 Especificación de los Escenarios de Calidad Relevantes 6](#_Toc179034664)

[4.4 Parte Estructural 9](#_Toc179034665)

[4.5 Parte Dinámica 9](#_Toc179034666)

[5 Vista de Procesos 10](#_Toc179034667)

[6 Vista de Implementación 11](#_Toc179034668)

[7 Vista de Despliegue 11](#_Toc179034669)

[8 Decisiones de Diseño y Selección de Alternativas 12](#_Toc179034670)

[9 Análisis de Reutilización 12](#_Toc179034671)

# Introducción

En el siguiente documento, identificamos aspectos de la arquitectura de software, las especificaciones planteadas en este archivo fueron desarrolladas siguiendo la metodología tradicional incremental; Además en este archivo realizamos el análisis 4+1 planteando las vistas pertinentes para el desarrollo e integración de la arquitectura de forma correcta.

## 

## Contexto del Problema y Propósito

El contextodelsistema SINH2O está relacionado con la necesidad de modernizar la gestión del cuidado de los cultivos para enfrentar desafíos como la escasez hídrica y la producción de cultivos de buena calidad con el menor uso posible de los recursos.

El propósito del sistema SINH2O es facilitar y modernizar la gestión de los cultivos mediante un sistema automatizado que utiliza sensores de humedad y controladores Arduino para administrar los recursos.

## Ámbito

El sistema contará con registros de humedad del suelo, localización de los dispositivos, apagado y encendido remoto, registros de temperatura de la tierra.

El sistema no protegerá la cosecha de las heladas o futuras plagas que puedan amenazarlas, tampoco contará con un sistema de actualizaciones online ya que este sistema es completamente independiente de la red.

Los beneficios del siguiente proyecto son el enorme ahorro hídrico, ya que, el siguiente sistema está diseñado para irrigar solo cuando sus parámetros (humedad) lo indiquen, esto poniéndola por encima de su competencia y disminuyendo los costos del consumo de agua.

## Definiciones, acrónimos y abreviaciones

|  |  |
| --- | --- |
| **ACRÓNIMO** | **DESCRIPCIÓN** |
| **SINH20** | Sistema de Irrigación por Nivel de Humedad de 2 Orientaciones |
| **SAD** | Software Architecture Design |
| **ERS** | Especificación de requerimientos de software |
| **R°** | Requerimiento |

## Referencias

A continuación, se listan las referencias a otros documentos:

* Casos de Uso
* Informe ERS
* Planilla de requerimientos

## Representación

La arquitectura del sistema del proyecto S.I.N.H.2.O está representada por los requerimientos funcionales y los no funcionales, además del mockup.

# Metas y Restricciones de la Arquitectura

A continuación, se revisan las metas y restricciones de la arquitectura.

## Metas de la arquitectura

A partir de las reuniones con los StakeHolders y del análisis de los requerimientos, se han definido las principales metas arquitectónicas del proyecto, las cuales servirán como directrices para el diseño y desarrollo del sistema:

* **Desempeño**:
  + Objetivo: alcanzar un 85% de efectividad en distribución de agua y control de cultivos.
  + Descripción: Busca optimizar el uso de los recursos hídricos y mejorar el manejo de los cultivos mediante la implementación de algoritmos inteligentes y control autorizado.
* **Tolerancia a fallos**:
  + Objetivos: Conseguir un 80% de tolerancia a fallos, teniendo en cuenta la posibilidad de errores no previstos durante la integración de dispositivos y el software.
  + Descripción: La capacidad del sistema para manejar y recuperarse de fallos es crucial para asegurar la continuidad operativa, minimizando el impacto de errores en los dispositivos.
* **Seguridad:** 
  + Objetivos: Lograr un 99% de protección de datos personales y confidenciales, garantizando la seguridad y privacidad de la información gestionada por el sistema.
  + Descripción: La seguridad se enfoca en salvaguardar los datos sensibles contra accesos no autorizados y otras amenazas potenciales.
* **Modificabilidad/Reusó**:
  + Objetivos: Diseñar un sistema altamente modificable y reutilizable para adaptarse a diferentes tipos de cultivos y escenarios agrícolas.
  + Descripción: Esto busca crear una arquitectura flexible que permite ajustes rápidos en la configuración y los algoritmos.

# Restricciones de la Arquitectura

Durante las reuniones con los StakeHolders, se identificaron varias restricciones que influyen en el diseño y desarrollo de la arquitectura del sistema. Estas restricciones son claves para la planificación y ejecución del proyecto:

* **Tiempo de construcción**: El proyecto debe completarse en un plazo de <?> meses abarcando todas las etapas de desarrollo, pruebas, implementación y capacitación del personal agrícola.
* **Infraestructura**: El sistema debe operar sobre servidores dedicados con balanceadores de carga para manejar las fluctuaciones de uso.
* **Otros componentes de software**: Se contempla la adquisición y licenciamiento de software adicional necesario para la integración.

## Otros antecedentes y consideraciones

* Factores ambientales y regulatorios:
  + El proyecto debe considerar normativas ambientales que regulen el uso del agua y el manejo de pesticidas y fertilizantes, además también debe seguir estándares agrícolas que aseguren el correcto uso de los dispositivos.
* Capacitación:
  + Es necesario contemplar programas de capacitación para el personal que operará el sistema, siendo este: formación en el uso de la interfaz de usuario, interpretación de datos generados y mantenimiento básico del hardware.
* Consideraciones financieras:
  + Es crucial manejar los recursos financieros con eficiencia para evitar sobrecostos y asegurar que los fondos sean suficientes para cubrir todas las fases del proyecto.

# Vista de Casos de Uso y Escenarios de Calidad

Esta sección describe en detalle el conjunto de escenarios funcionales y no funcionales que obtuvieron la mayor prioridad en el análisis. Para esto se presenta y describe el diagrama de casos de uso y los casos de uso prioritarios, así como los escenarios en que uno o más atributos de calidad se ven involucrados de manera significativa.

## Modelo de Casos de Uso

El modelo de casos de uso puede ser encontrado en la carpeta “Diagrama casos de uso”, “Casos de uso extendido”.

*Google Drive: Sign-in*. (s. f.). [Diagrama casos de uso](https://drive.google.com/drive/u/0/folders/1fRNixx9vXbUnvSynCC2JxMnlO19vtyDt)

*Google Drive: Sign-in*. (s. f.-b). [Casos de uso extendido](https://drive.google.com/drive/u/0/folders/1sxZ-QcE7vbwxhGrVqo0TaxlqWbj6QDvR)

## Especificación de Casos de Uso Relevantes

Los casos de uso considerados los más relevantes para el desarrollo de la arquitectura fueron determinados, según los siguientes criterios:

* Su implementación implica varios nodos de la vista de despliegue.
* Su implementación es de alto riesgo.
* Incluye muchos conceptos y relaciones del dominio.
* Incluye posibles escenarios críticos de calidad.

A continuación, se listan los casos de uso relevantes, los cuales pueden ser encontrados con su especificación detallada en el documento “Casos de Uso extendido”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Código** | **Nombre** | **Actores** | **Prioridad** |
| **R°1** | Registro de humedad de la tierra | Usuario Administrador | Alta |
| **R°5** | Localización del dispositivo | Usuario Administrador | Alta |
| **R°6** | Apagado y encendido remoto | Usuario Administrador | Alta |
| **R°15** | Configuración remota | Usuario Administrador | Muy Alta |
| **R°16** | Apagado de emergencia | Sistema | Muy Alta |
| **R°21** | Detección de obstrucciones | Sistema | Alta |
| **R°14** | Integración y eliminación de sensores | Usuario Administrador | Media |
| **R°17** | Historial de recarga de agua | Usuarios Administrador y Usuario | Media |
| **R°12** | Humedad a tiempo real | Usuarios Administrador y Usuario | Media |
| **R°13** | Estado del clima | Sistema | Media |

## Especificación de los Escenarios de Calidad Relevantes

Después de un análisis en conjunto con los StakeHolders, los escenarios de calidad se expresan a continuación:

**ID: R°1**

**Nombre:** Registro de humedad de la tierra

**Sinopsis:** El tiempo de respuesta para registrar y mostrar la humedad del suelo en intervalos definidos debe ser eficiente para el monitoreo.

**Entorno:** Proceso normal de operación con varios dispositivos conectados.

**Cambio en el entorno:** Monitoreo de múltiples cuadrantes en diferentes módulos.

**Comportamiento esperado:** La humedad se muestra en tiempo real sin retrasos perceptibles.

**Medida:** El tiempo de respuesta debe ser menor a 2 segundos por cuadrante.

**Prioridad Arquitectónica:** Alta

**Aplicación:** Global

**ID: R°5**

**Nombre:** Localización del dispositivo  
**Sinopsis:** La localización del dispositivo debe ser visible en tiempo real, sin interrupciones en la visualización.  
**Entorno:** Proceso normal de operación con varios dispositivos conectados.  
**Cambio en el entorno:** Pérdida de conexión o fallos en la red.  
**Comportamiento esperado:** El sistema recupera la localización automáticamente tras una interrupción.  
**Medida:** El tiempo de recuperación debe ser menor a 5 segundos después de la reconexión.  
**Prioridad Arquitectónica:** Alta  
**Aplicación:** Global

**ID: R°6**

**Nombre:** Apagado y encendido remoto

**Sinopsis:** El sistema debe permitir que los dispositivos se apaguen o enciendan de forma remota de manera segura y eficaz.

**Entorno:** Proceso de operación bajo credenciales de administrador.

**Cambio en el entorno:** Intentos de acceso no autorizado o fallo en la red.

**Comportamiento esperado:** Solo se responden solicitudes autorizadas para el apagado y encendido remoto.

**Medida:** El tiempo de ejecución de la acción debe ser menor a 3 segundos, y el 100% de solicitudes no autorizadas debe ser rechazado.

**Prioridad Arquitectónica:** Alta

**Aplicación:** Global

**ID: R°15**

**Nombre:** Configuración remota

**Sinopsis:** Los administradores deben poder modificar parámetros del sistema de manera remota sin afectar la estabilidad del sistema.

**Entorno:** Proceso normal de operación con varios dispositivos activos.

**Cambio en el entorno:** Configuración simultánea de múltiples dispositivos.

**Comportamiento esperado:** Los cambios se aplican sin interrumpir la operación del sistema.

**Medida:** El sistema debe aplicar los cambios en menos de 10 segundos y sin impacto en la operación en curso.

**Prioridad Arquitectónica:** Muy Alta

**Aplicación:** Global

**ID: R°16**

**Nombre:** Apagado de emergencia

**Sinopsis:** El sistema debe ejecutar un apagado de emergencia en caso de fallo, para evitar daños mayores en el equipo o la operación.

**Entorno:** Evento crítico o fallo en los sensores del sistema.

**Cambio en el entorno:** Detectar una sobrecarga o fallo crítico en los componentes.

**Comportamiento esperado:** El sistema debe ejecutar un apagado automático y registrar el evento de emergencia.

**Medida:** El apagado debe realizarse en menos de 1 segundo tras detectar el fallo.

**Prioridad Arquitectónica:** Muy Alta

**Aplicación:** Global

**ID: R°21**

**Nombre:** Detección de obstrucciones

**Sinopsis:** El sistema debe identificar obstrucciones en el flujo de agua del sistema de riego y alertar al administrador.

**Entorno:** Proceso normal de riego con varios dispositivos activos.

**Cambio en el entorno:** Flujo irregular o bloqueado en los sistemas de riego.

**Comportamiento esperado:** El sistema detecta la obstrucción y envía una alerta al administrador.

**Medida:** La detección debe realizarse en menos de 1 minuto después del inicio de la obstrucción.

**Prioridad Arquitectónica:** Alta

**Aplicación:** Global

**ID: R°14**

**Nombre:** Integración y eliminación de sensores

**Sinopsis:** El sistema debe permitir la gestión de sensores, añadiendo o eliminando estos sin afectar la operación.

**Entorno:** Proceso normal con múltiples sensores operando.

**Cambio en el entorno:** Añadir o eliminar sensores mientras el sistema está en uso.

**Comportamiento esperado:** El sistema reconoce y gestiona los sensores sin interrupciones en el monitoreo.

**Medida:** La integración o eliminación debe completarse en menos de 5 minutos sin afectar el desempeño del sistema.

**Prioridad Arquitectónica:** Media

**Aplicación:** Global

**ID: R°17**

**Nombre:** Historial de recarga de agua  
**Sinopsis:** El sistema debe almacenar y mostrar el historial de recargas de los estanques de agua con detalles sobre la fecha y hora.  
**Entorno:** Operación normal con varios estanques en funcionamiento.  
**Cambio en el entorno:** Solicitudes simultáneas para acceder al historial.  
**Comportamiento esperado:** Los usuarios pueden acceder al historial sin retrasos o fallos en la visualización.  
**Medida:** El historial debe cargarse en menos de 3 segundos para el último mes de registros.  
**Prioridad Arquitectónica:** Media  
**Aplicación:** Global

**ID: R°12**

**Nombre:** Humedad a tiempo real

**Sinopsis:** El sistema debe mostrar el nivel de humedad en tiempo real para todos los cuadrantes monitorizados.

**Entorno:** Monitoreo continuo con múltiples cuadrantes activos.

**Cambio en el entorno:** Variaciones en los niveles de humedad debido a factores externos.

**Comportamiento esperado:** El sistema actualiza los datos de humedad en tiempo real sin retrasos perceptibles.

**Medida:** La actualización debe ocurrir cada 5 segundos.

**Prioridad Arquitectónica:** Media

**Aplicación:** Global

**ID: R°13**

**Nombre:** Estado del clima

**Sinopsis:** El sistema debe proporcionar información precisa sobre el clima en tiempo real.

**Entorno:** Proceso normal de operación con monitoreo climático activo.

**Cambio en el entorno:** Cambios bruscos en las condiciones climáticas.

**Comportamiento esperado:** El sistema refleja los cambios de clima en tiempo real sin interrupciones.

**Medida:** El sistema debe actualizar el estado del clima cada 10 minutos.

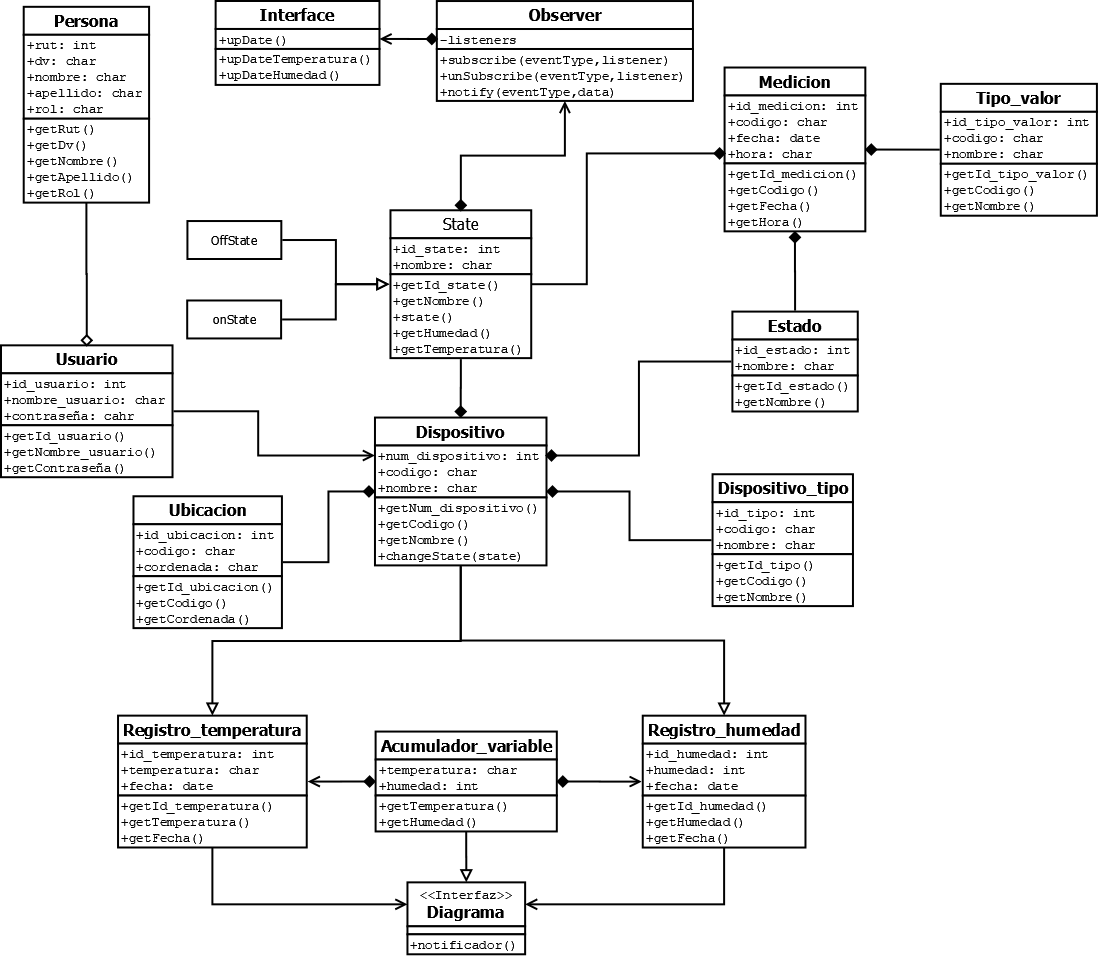
**Prioridad Arquitectónica:** Media

**Aplicación:** Global

**Vista Lógica**

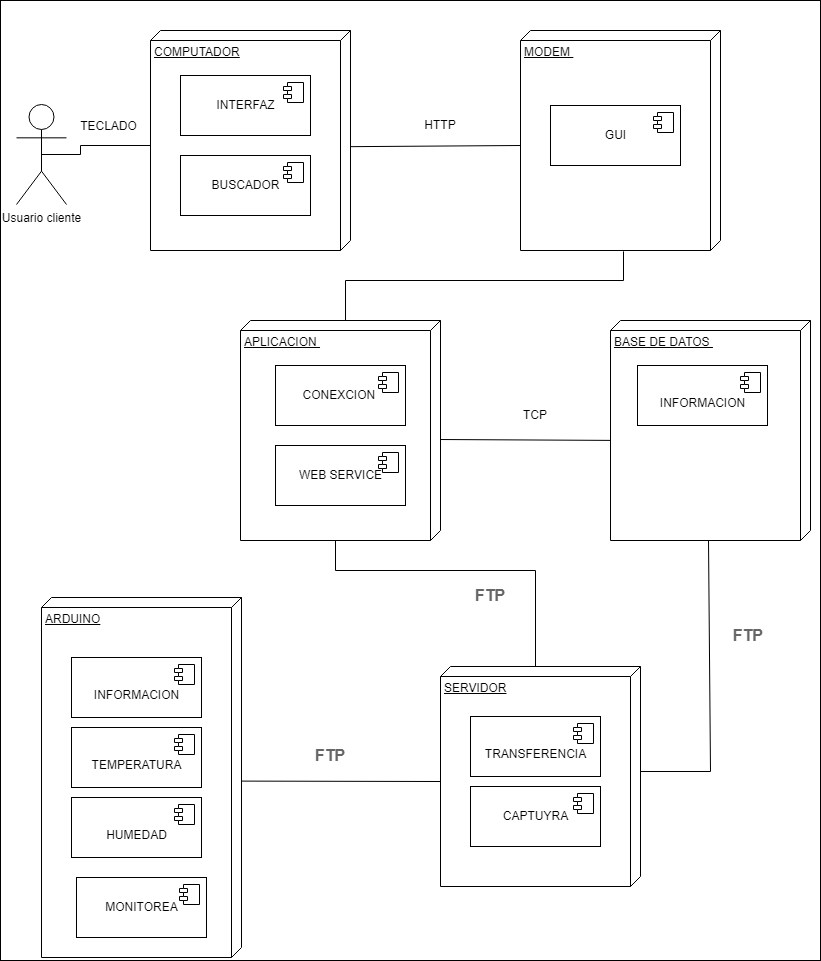
En el siguiente apartado se mostrará una vista lógica de la aplicación, expresadas en dos diagramas uno de ellos que muestra la parte estática (módulos), y otra vista que representa la parte dinámica (componentes y conectores).

## Parte Estructural

Ilustración 1: Diagrama de Módulos

## Parte Dinámica

Ilustración 2: Diagrama de Componentes y Conectores

****

# Vista de Procesos

A continuación, se muestra una vista de procesos, en la cual se observa que:

Estos pueden ser encontrados en la carpeta “Diagramas de actividad”, “Diagramas de secuencia”.

*Google Drive: Sign-in*. (s. f.-c). [Diagramas de actividad](https://drive.google.com/drive/u/0/folders/1HcB2B5qubwNn2txuTSHOqUdBbxIVw8Bt)

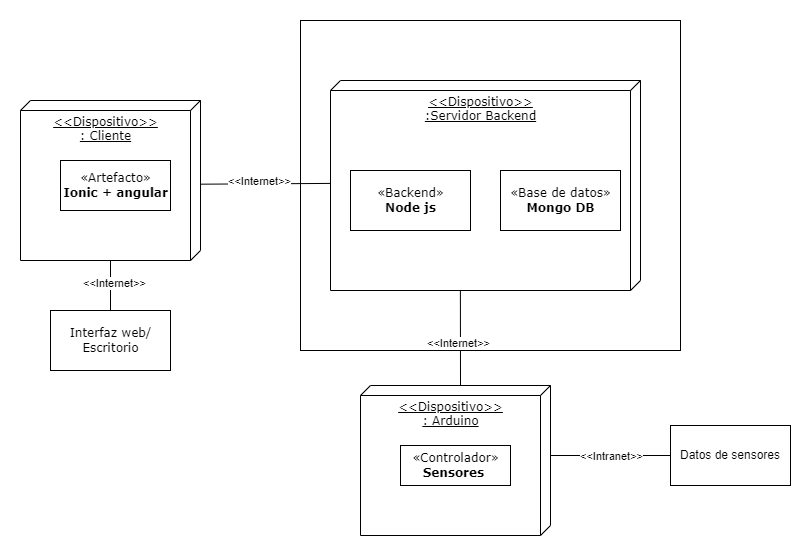
*Google Drive: Sign-in*. (s. f.-d). [Diagramas de secuencia](https://drive.google.com/drive/u/0/folders/16W9usLXO_HB5cvWs_rcbmEGS96QipQXe)

# Vista de Implementación

En esta vista se aprecia los módulos principales que contendrán distintas funcionalidades de la aplicación:

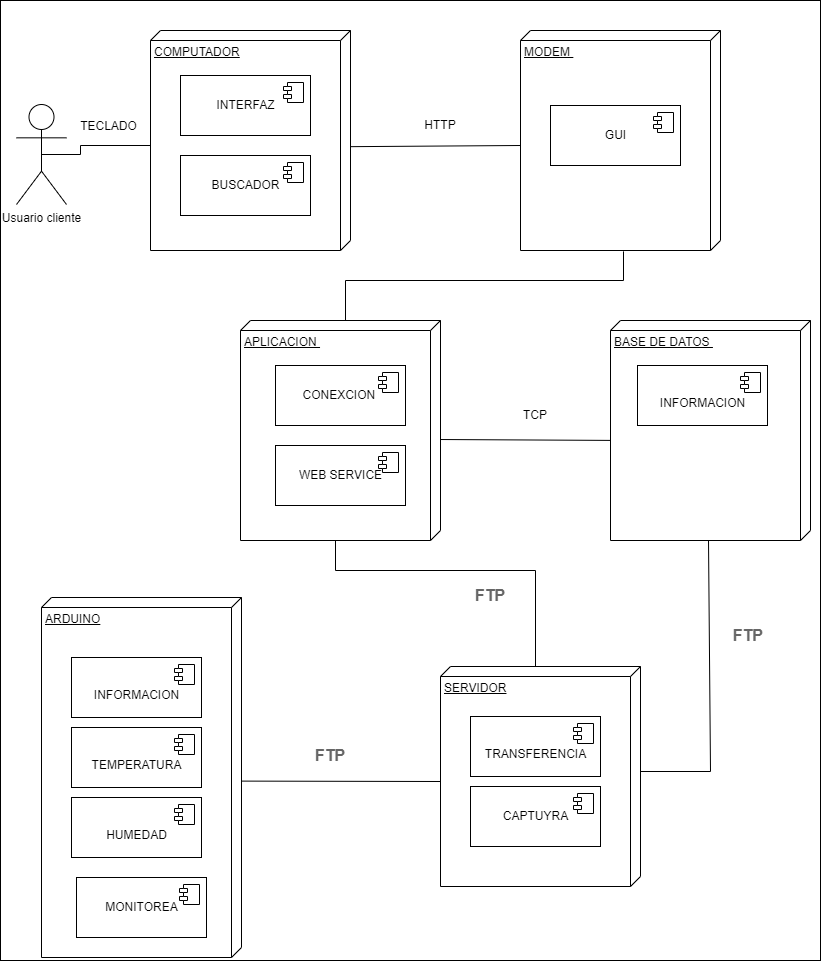
*Mockup SINH2O*. (s. f.). Figma. https://www.figma.com/design/aR30yILk8FMSfU59JDBRxZ/Mockup-SINH2O?node-id=0-1&node-type=canvas&t=ivCugMukt3Hpf2yu-0

Ilustración 4: Vista de Implementación



# Vista de Despliegue

En esta vista se despliegan los nodos que participan con el sistema. Los nodos principales son los nodos Servidor de Integración. Características a continuación:

****

# Decisiones de Diseño y Selección de Alternativas

Las decisiones arquitectónicas se basaron en las restricciones de Tiempo de Construcción y Recursos Disponibles. Dado que el proyecto SINH2O debe estar operativo en un periodo limitado, se decidió optar por una arquitectura basada en ionic con angular para el frontend y node.js en el backend, aprovechando el conocimiento previo del equipo y las características de estas tecnologías.

Se decidió modularizar la arquitectura dividiendo el sistema en varios componentes claramente definidos, tales como el frontend (interfaz web y de escritorio), el backend (API REST y lógica de negocio), y la integración con los dispositivos de hardware (Arduino y sensores).

# Análisis de Reutilización

En el desarrollo del proyecto SINH2O, se decidió maximizar la reutilización de componentes de software ya probados para reducir el tiempo de desarrollo y minimizar errores potenciales:

* Reutilización de Componentes de Frontend: Se decidió utilizar Ionic Framework en combinación con Angular, lo que permite reutilizar una gran cantidad de componentes predefinidos para la interfaz de usuario, como formularios, gráficos y listas interactivas. Estos componentes ya están optimizados para aplicaciones web y de escritorio, y solo requieren personalización en términos de estilo y adaptación a los requisitos específicos del sistema.
* Reutilización de Backend: En el backend, se optó por usar Node.js con Express.js, que ofrece una estructura ligera y permite integrar fácilmente bibliotecas de terceros. En cuanto a la base de datos, MongoDB fue seleccionada debido a su flexibilidad y capacidad de escalar sin problemas.