



Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos basados en el Internet de las Cosas

Presentado por:

Br. Pedro Boll

CI: 20.173.376

Tutor:

Prof. Antonio Russoniello

Agenda

- Introducción.
- Planteamiento del Problema
 - Análisis del Problema.
 - Objetivos del Trabajo Especial de Grado.
 - Alcance.
- Marco Teórico
 - Internet de las Cosas.
 - Herramientas de Visualización, Monitoreo y Control.
 - Placas Programables.
- Marco Metodológico y Herramientas de Desarrollo
 - Diseño e Implementación.
 - Herramientas de Desarrollo.
 - Entorno de Pruebas



Agenda

- Marco Aplicativo
 - Prototipos de Dispositivos IoT.
 - Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos.
 - Resultados.
- Conclusiones
- Trabajos Futuros
- Referencias



Introducción



Introducción

- Día tras día, la cantidad de dispositivos que se conectan a internet se incrementa en cantidad y en variedad. Los artefactos que ahora poseen capacidades extendidas mas allá de su especificación original gracias a la presencia de sensores, actuadores.
- El volumen de datos que se generan a partir de ellos crea dificultades para analizarlos u observarlos en tiempo real. La capacidad de controlarlos a todos de manera adecuada y de generar procesos automatizados también se vuelve una tarea compleja.
- Se ha diseñado y creado una alternativa de software que permite centralizar la observación de datos y el control de dispositivos, utilizando un escenario realista con prototipos funcionales.



Planteamiento del Problema



Planteamiento del Problema

Análisis del Problema

- Es común ver artefactos con capacidades extendidas de las que originalmente fueron diseñados. La posibilidad de medir variables contextuales gracias a sensores o de actuar sobre el ambiente con actuadores, junto con la capacidad de conectarse a internet hacen que dichos dispositivos denominados “inteligentes” puedan llevar a cabo tareas de manera muy eficiente.
- Sin embargo la masificación de este tipo de tecnología ha traído retos en lo que respecta a la observabilidad de las operaciones y la información asociada a dichos dispositivos.



Planteamiento del Problema

Objetivo del Trabajo Especial de Grado

Se busca proveer un software que pueda ofrecer la capacidad de observación de datos tanto en tiempo real, como histórica de sensores de dispositivos inteligentes así como también la posibilidad de monitorear el estado actual de los mismos y poder controlarlos de manera centralizada y flexible, bajo el marco de un escenario realista haciendo uso de prototipos funcionales de dispositivos basados en el concepto del internet de las cosas



Planteamiento del Problema

Objetivos Específicos del Trabajo Especial de Grado

- Construir prototipos funcionales de dispositivos IoT para la captura de datos y que sean capaces de modificar su ambiente.
- Establecer una arquitectura para una solución modular para un sistema visualización de datos y monitoreo y control de dispositivos IoT.
- Seleccionar y utilizar herramientas de código libre de cara a la capacidad de observabilidad, control y automatización requeridos.
- Crear una interfaz donde se puedan acceder de manera centralizada las herramientas seleccionadas cuya integración permita explotar las ventajas de cada una.
- Validar con los prototipos el funcionamiento transversal de la solución.



Planteamiento del Problema

Alcance (Prototipos de Dispositivos IoT)

Por el lado de los prototipos de dispositivos, estos deben cumplir lo siguiente:

- Poder obtener datos reales sobre variables ambientales a través del uso de sensores.
- Realizar tareas que pueda alterar de alguna forma el ambiente haciendo uso de actuadores.
- Posibilitar la capacidad de despliegue de los dispositivos en diferentes ambientes o distintas tareas.
- Conectarse entre ellos usando algún protocolo o estándar libre de comunicación.



Planteamiento del Problema

Alcance (Software de Visualización, Monitoreo y Control)

El sistema planteado debe tener la capacidad de:

- Permitir la observación de datos en tiempo real e históricos de dispositivos.
- Posibilitar el monitoreo del estado actual de los sensores/actuadores presentes en los dispositivos.
- Realizar el control (manual o automatizado) de tareas de los dispositivos conectados.
- Tener acceso a dichas capacidades de forma centralizada.



Marco Teórico



Internet de las Cosas

Definición

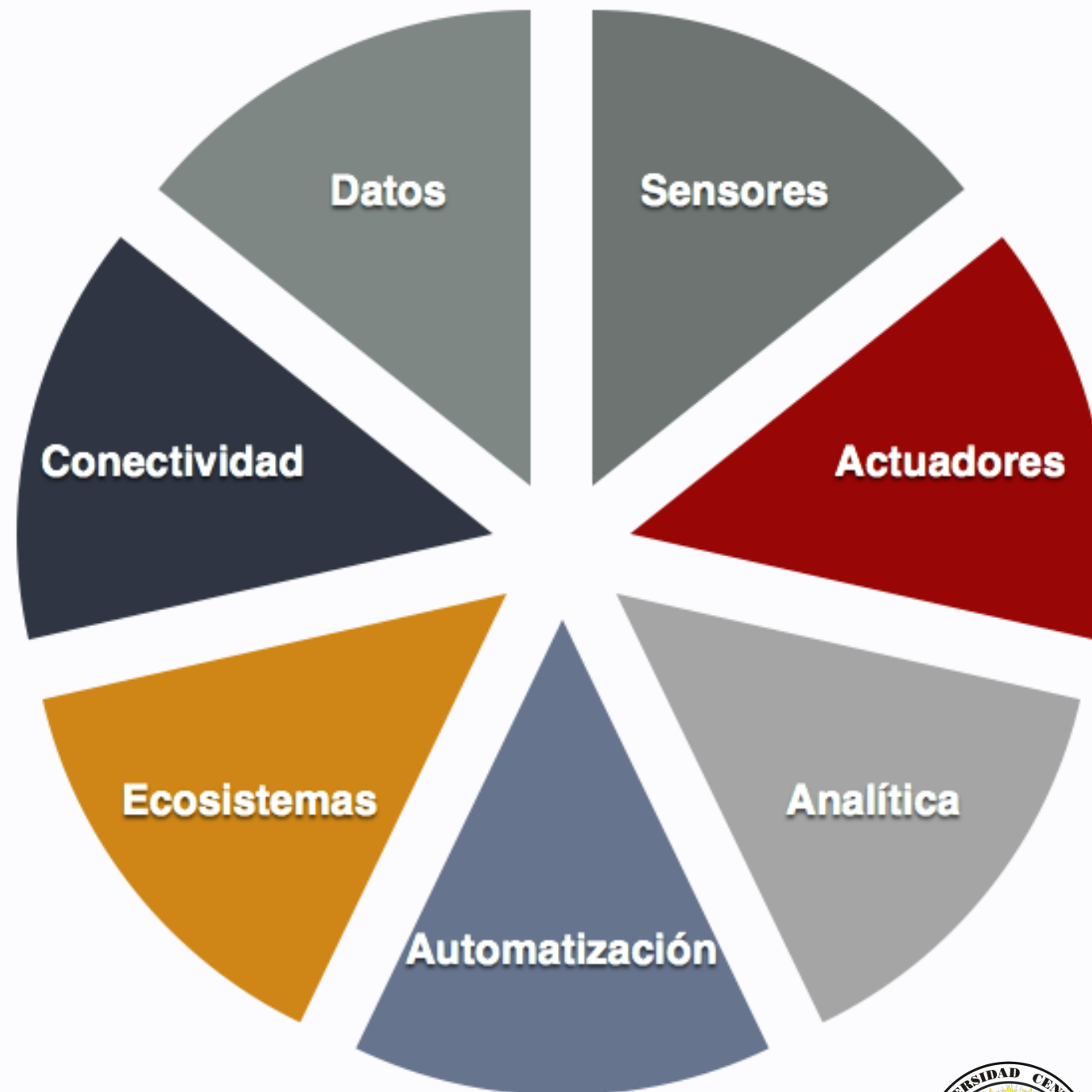
El termino internet de las cosas engloba tanto a los dispositivos embebidos que utilizan servicios de comunicación ofrecidos por los protocolos de internet [1] como al marco de desarrollo en el que todas las cosas tienen una representación y una presencia en internet [2].

Aquellos dispositivos y objetos cotidianos con capacidad de comunicarse entre si y con otras plataformas, con un poder de computo en crecimiento continuo y dotados de sensores y actuadores forman ya parte una revolución en cuanto al uso de la tecnología.



Internet de las Cosas

Características de los dispositivos IoT



Herramientas de Visualización, Monitoreo y Control

Herramientas de Visualización

Los seres humanos podemos comprender mucho más rápidamente la información si somos capaces de visualizarla. Las herramientas que ofrecen esa capacidad de organizarla son llamadas herramientas de visualización.

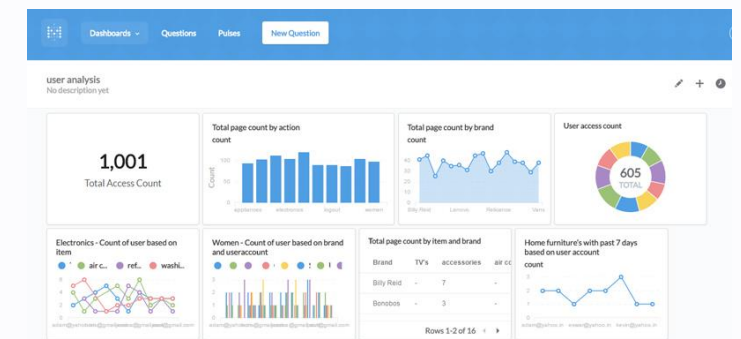
Ofrecen amplia gama de modos de presentar datos, dependiendo del tipo y el objetivo que se busca al mostrarlos.



Grafana



Tableau



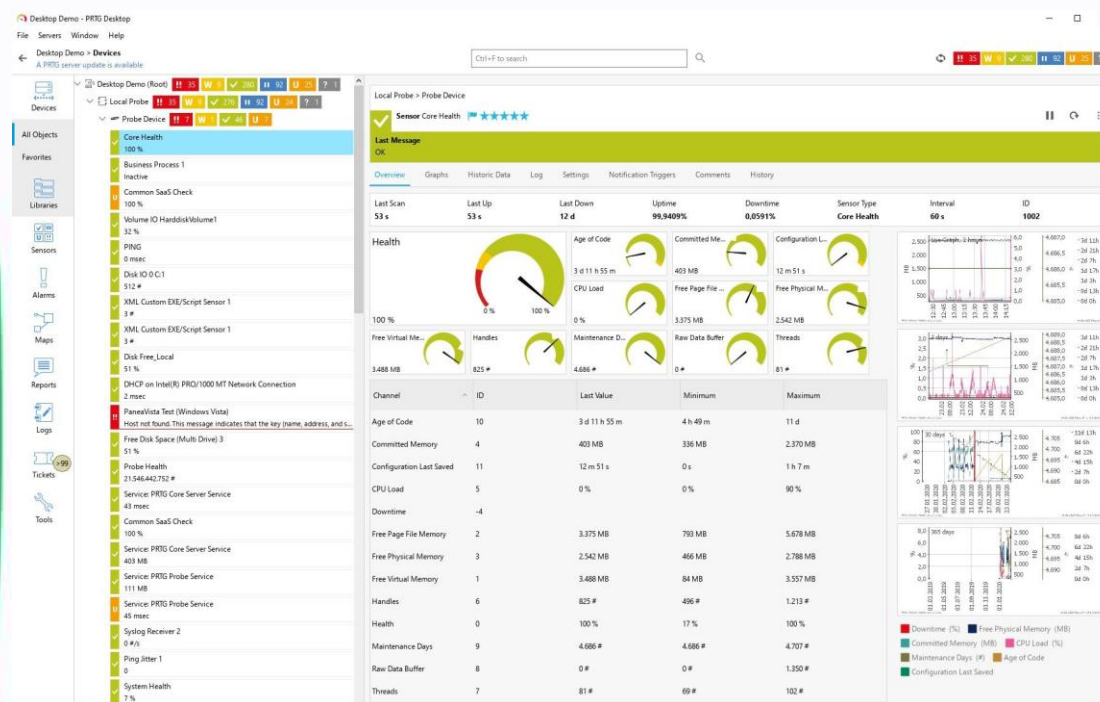
Metabase



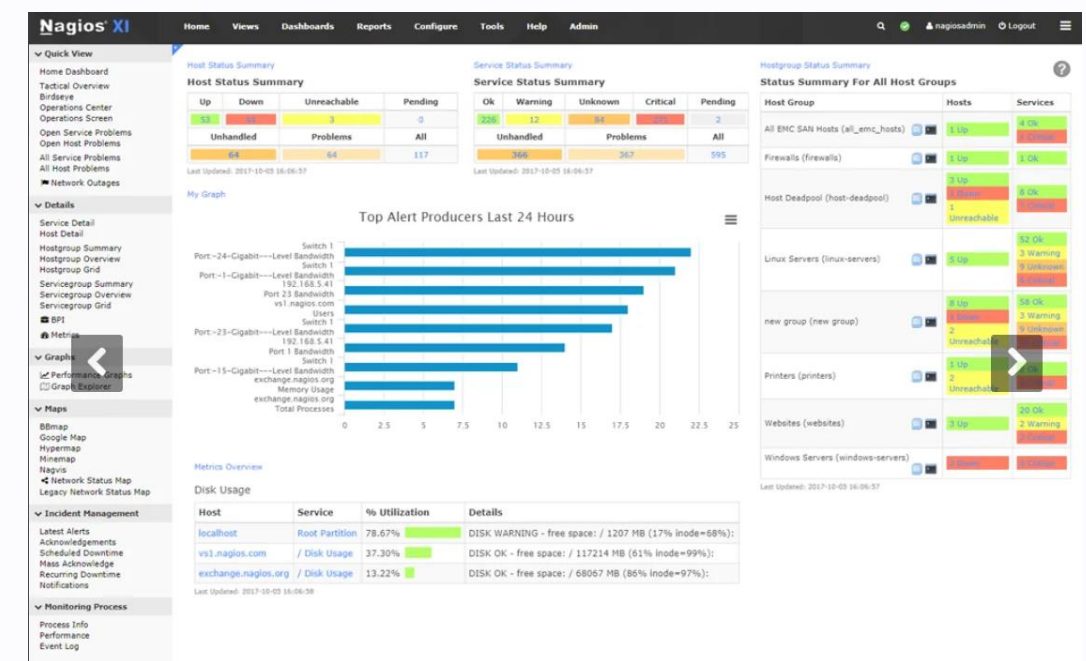
Herramientas de Visualización, Monitoreo y Control

Herramientas de Monitoreo

El monitoreo de procesos, componentes, artefactos o dispositivos no es algo nuevo. Dependiendo del área podemos encontrar multitud de herramientas que permiten realizar esto.



PRTG



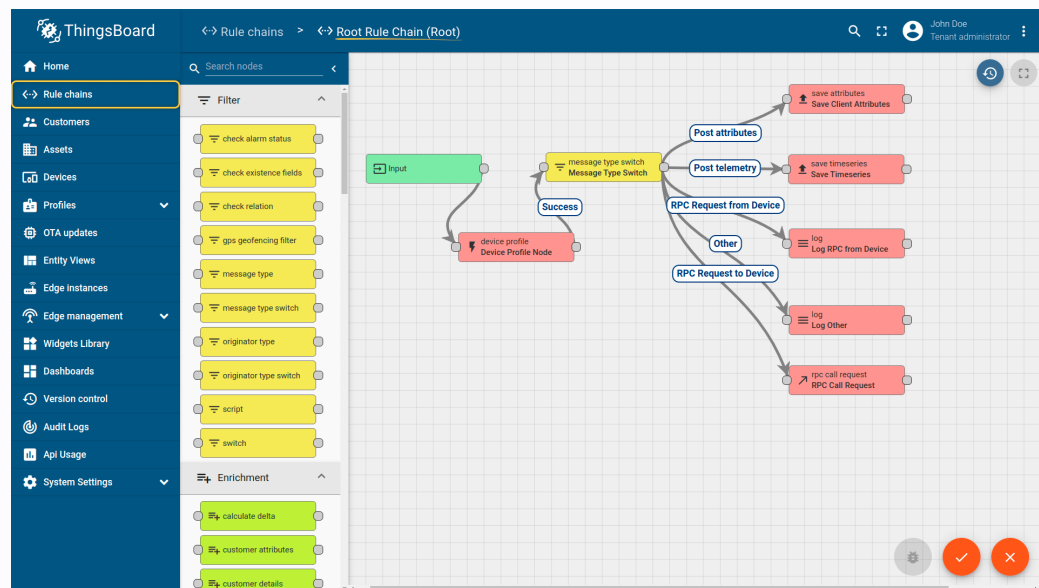
Nagios



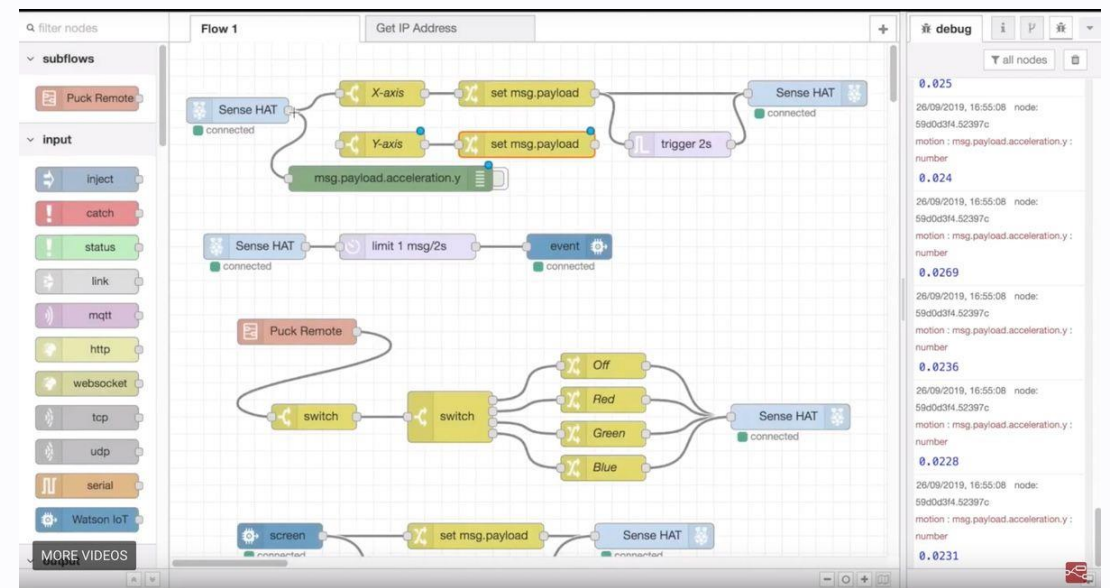
Herramientas de Visualización, Monitoreo y Control

Herramientas de Control

En el caso del control de dispositivos y la automatización de tareas, una herramienta que sea capaz de ser flexible ante las diferentes arquitecturas, protocolos y estándares que pueden presentar los artefactos es indispensable, pero a su vez las hacen herramientas complejas de uso y en general se requiere cierto conocimiento técnico para manejarlas.



ThinkBoard



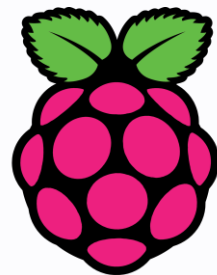
Node-Red



Placas Programables

¿Qué son las placas programables?

Las placas programables son microcontroladores y microcomputadores utilizados para realizar prototipos de estos dispositivos a un bajo costo más conocidas son placas Arduino y o SoC como los Raspberry Pi, creados bajo la idea de ser hardware abierto y con los cuales se puede probar la electrónica básica, sensores y actuadores necesarios para automatizar procesos.



RaspberryPi



Marco Metodológico y Herramientas de Desarrollo

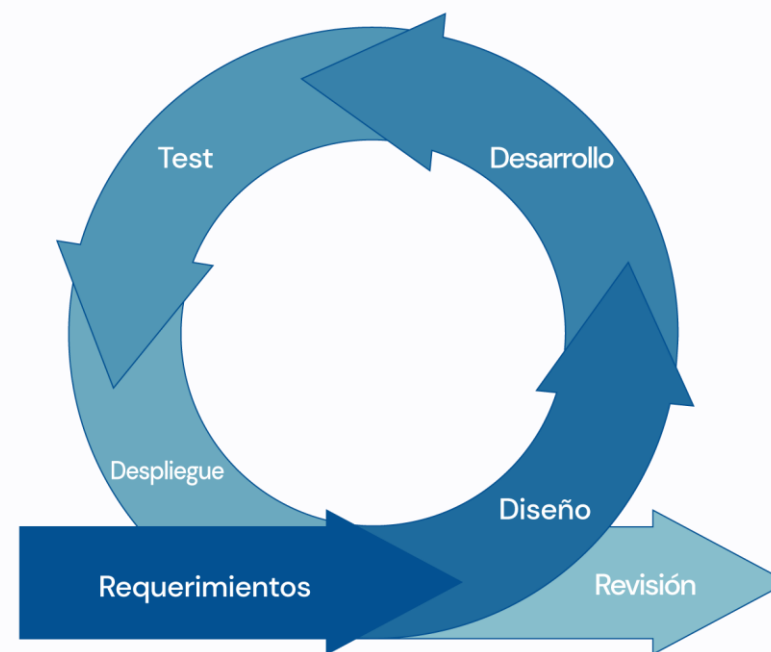


Diseño e Implementación

Etapa de Implementación

Para la implementación se uso una metodología ágil en la que se plantearon tareas dentro de sprints de dos semanas de duración donde se busco realizar entregables incrementables.

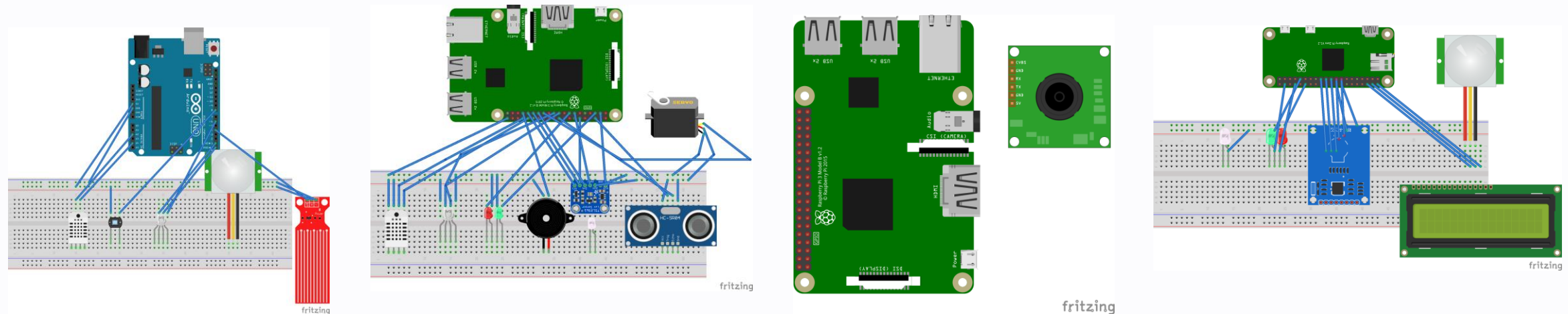
En su totalidad el desarrollo termino llevando tres sprints para el desarrollo de los prototipos funcionales en los dispositivos IoT y cuatro sprints para el desarrollo de la aplicación web.



Diseño e Implementación

Etapa de Diseño de dispositivos

El diseño de se comenzó por los prototipos funcionales de dispositivos IoT. En el diseño se busco establecer el uso de diferentes enfoques tecnológicos (microcontroladores y microcomputadores) y sus componentes (sensores diferentes para medir las mismas variables, actuadores para cambiar el ambiente). Por el lado de las comunicaciones y para simplificar se planteo el uso de un solo tipo de protocolo.



Diseño e Implementación

Etapa de Implementación de dispositivos

Se crearon cuatro prototipos de dispositivos, dos de ellos con tipo de sensores espejo para poder medir la diferencias en los datos reportados por sensores en dos ubicaciones distintas y dos sensores dedicados a la seguridad, uno dedicado a sensores y actuadores para acceso y uno para hacer de cámara de seguridad inteligente.



Diseño e Implementación

Etapa de Diseño aplicación web

En el caso de la aplicación web, la meta fue ser capaz de centralizar las herramientas de visualización de datos y de monitoreo y control de dispositivos de forma embebida.

También se tuvo en cuenta una estructura monolítica basada en módulos de manera que pueda crecer de manera orgánica en términos de tecnologías y protocolos a implementar.



Diseño e Implementación

Etapa de Diseño aplicación web

Para implementar la arquitectura anterior se acordó el uso de tecnologías de contenedores para las aplicaciones de control, visualización y monitoreo elegidas para ser integradas dentro de la interfaz web.

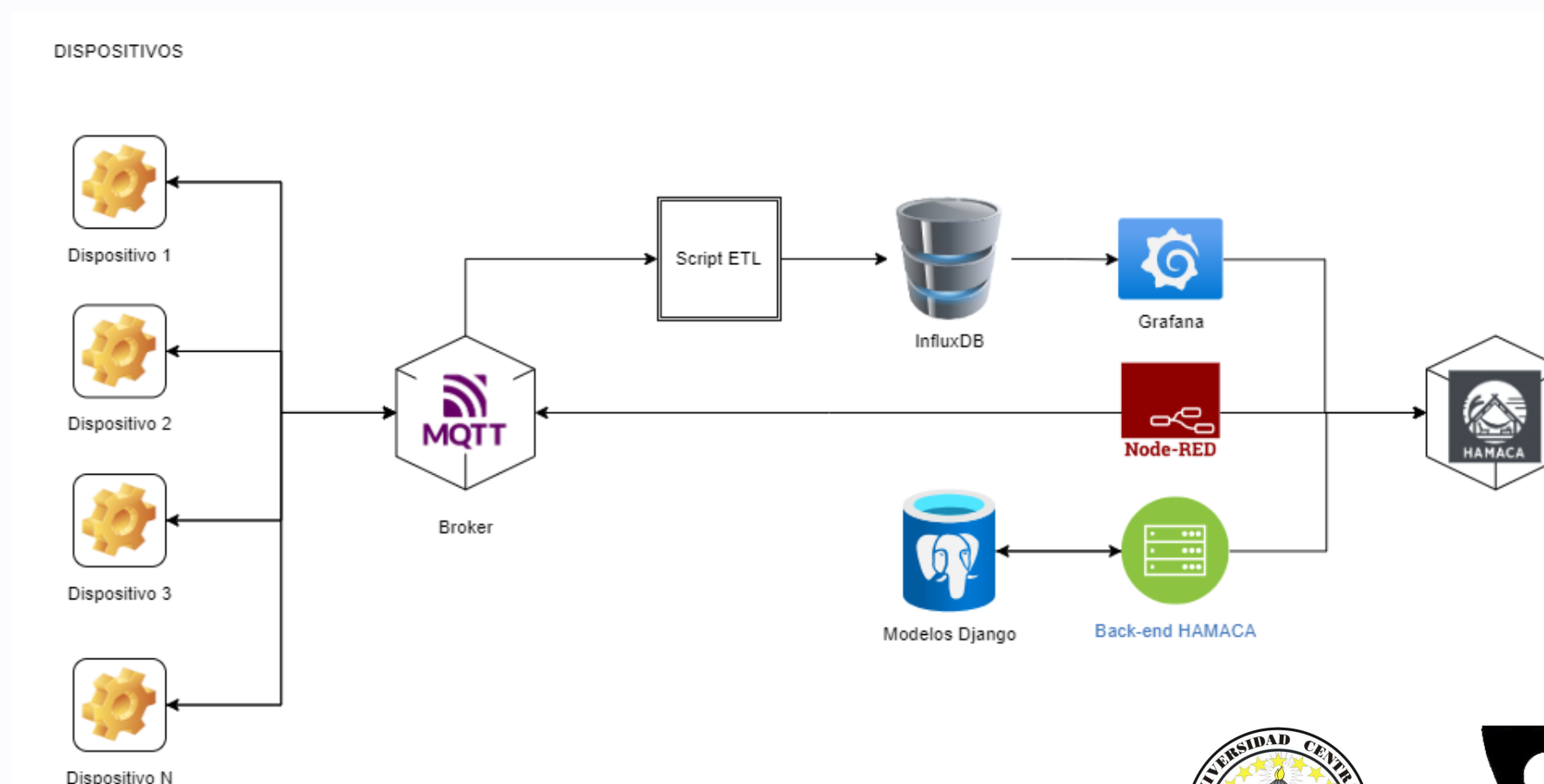
Para el almacenamiento de información en todo el sistema se definió usar dos bases de datos: una base de datos para los elementos del sistema web y otra con la tarea de almacenar los datos de los sensores. Finalmente un intermediario que controle la comunicación entre los dispositivos y el sistema.



Diseño e Implementación

Etapa de Implementación

Para la visualización de datos la herramienta seleccionada fue Grafana un software de código abierto capaz de crear cuadros de mando con gráficos. En el caso de la herramienta de control se escogió la herramienta de automatización de tareas Node-Red. Para la comunicación de los dispositivos y el sistema web se escogió el protocolo MQTT.



Herramientas de Desarrollo

Herramientas de Hardware

Para el caso de las herramientas de hardware se conto con un microcontrolador Arduino Uno. Fue programada bajo el lenguaje de programación de Arduino para poder comunicarse con los sensores y actuadores.

En el resto de los dispositivos se hizo uso de microcomputadores Raspberry Pi: dos del tipo modelo 3B y dos del tipo Zero. Al ser computadores estos usaron un derivado del sistema operativo Debian llamado Raspberry Pi OS (otrora Raspbian). Para programar el comportamiento de los sensores y actuadores se crearon varios scripts en el lenguaje de programación Python



Herramientas de Desarrollo

Herramientas de Software

La aplicación web fue creada en el lenguaje de programación Python bajo el framework Django. La base de datos para administración del sistema web escogido fue PostgreSQL, por su facilidad de uso y robustez. Para los datos provenientes de sensores y actuadores se optó por utilizar la base de datos NoSQL InfluxDB basada en series de tiempo.

Con respecto a las aplicaciones integradas al sistema, éstas fueron desplegadas haciendo uso de contenedores Docker orquestados a través de la tecnología Docker Compose.

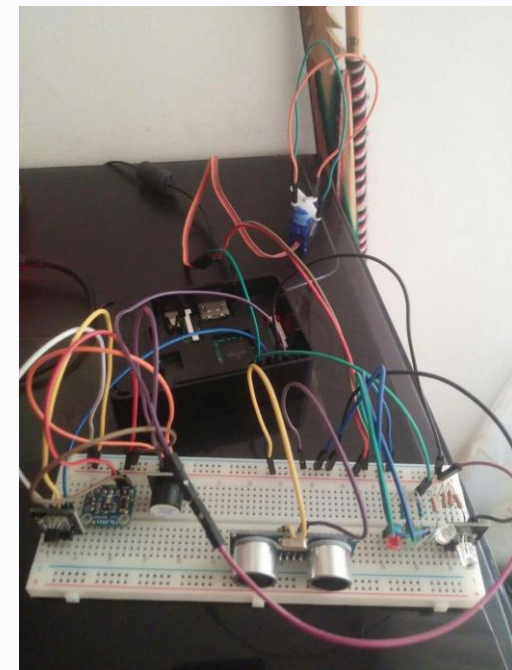
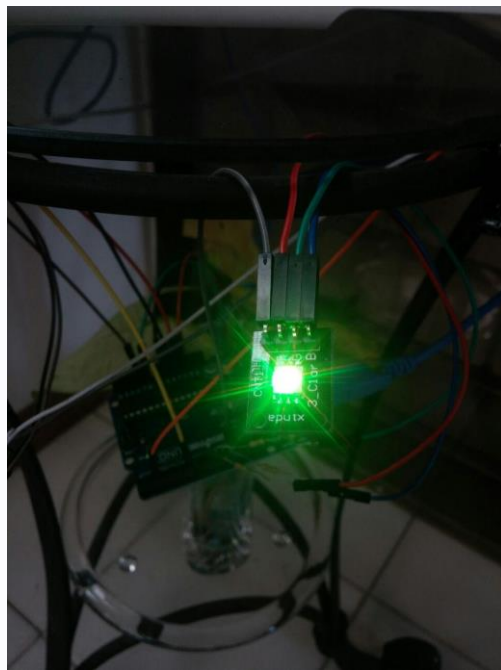
Por último, la comunicación fue a cargo de la implementación Mosquitto del protocolo MQTT.



Entornos de Prueba

Entorno de prueba para prototipos de dispositivos

Se armaron cuatro dispositivos distintos: dos con tareas de monitoreo de variables ambientales y dos dedicados a tareas de vigilancia y seguridad. Uno de cada categoría fue colocado en el interior y en el exterior del hogar para comparar las diferencias entre entornos.



Entornos de Prueba

Entorno de prueba para software HAMACA

La aplicación web fue desplegada en un Raspberry Pi Modelo 3B que fungió de punto de acceso WiFi para los dispositivos. La aplicación Integró un flujo de control, un dashboard de visualización y una interfaz de control para los dispositivos. Contó con el uso de un usuario administrador de sistema, así como un usuario con solo roles de visualización de las interfaces.

Se comprobó que cada herramienta a integrar pudiese cumplir con el proposito por la que fue elegida de manera individual y luego se hizo una prueba end to end durante un periodo de cinco meses con la integración realizada en los puntos de interés anteriormente mencionados.



Marco Aplicativo



Cronograma de actividades

Actividades establecidas para el trabajo de investigación

Actividad	Tiempo Estimado	Sprint
Levantamiento de requerimientos	Una semana	Sprint 1
Instalación y despliegue de Infraestructura	Una semana	Sprint 1
Diseño y prototipado de dispositivos	Una semana	Sprint 2
Desarrollo de scripts para sensores, actuadores y comunicación de datos	Una semana	Sprint 2 y sprint 3
Desarrollo de back-end del proceso de obtención de datos	Una semana	Sprint 3
Captura de datos de dispositivos	Doce semanas	Sprint 4 a sprint 10
Documentación de proceso de obtención de datos	Una semana	Sprint 4



Cronograma de actividades

Actividades establecidas para el trabajo de investigación

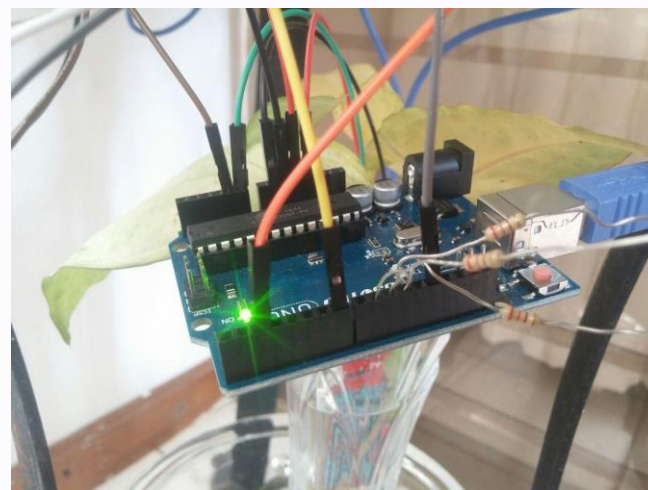
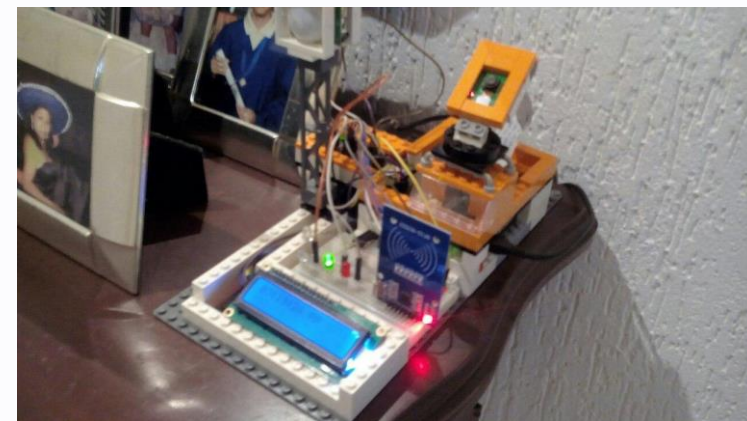
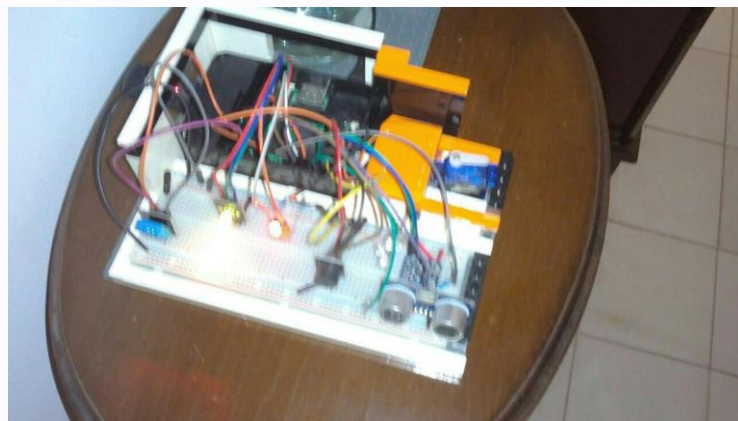
Actividad	Tiempo Estimado	Sprint
Implementación base de aplicación web	Una semana	Sprint 5
Desarrollo de módulo de gestión de aplicación	Una semana	Sprint 5
Desarrollo de módulo de visualización de datos	Una semana	Sprint 6
Desarrollo de módulo de control y automatización de dispositivos	Una semana	Sprint 6
Desarrollo de módulo de redes y comunicación de datos	Una semana	Sprint 7
Documentación de desarrollo de aplicación web	Dos semanas	Sprint 7 y Sprint 8
Pruebas de end to end de todos los flujos	Tres semanas	Sprint 8 y sprint 9
Recolección de resultados del proyecto de investigación	Dos semanas	Sprint 10



Prototipos de Dispositivos IoT

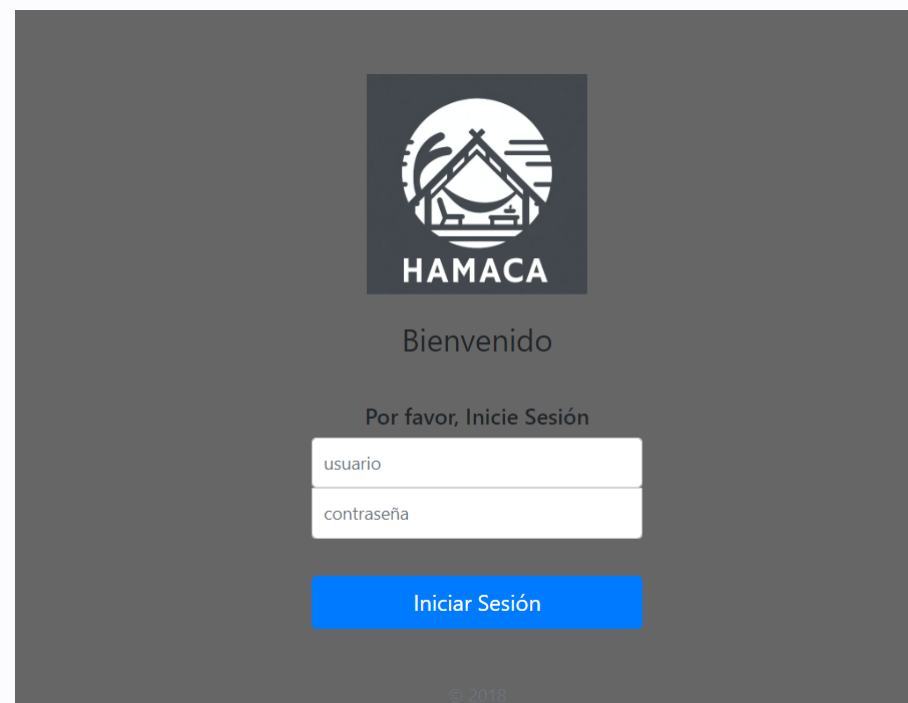
¿Qué prototipos fueron creados?

Se crearon cuatro prototipos funcionales de dispositivos IoT. Dos de ellos en equidad de tipo de sensores (ambientales) y dos enfocados en el tópico de la seguridad.



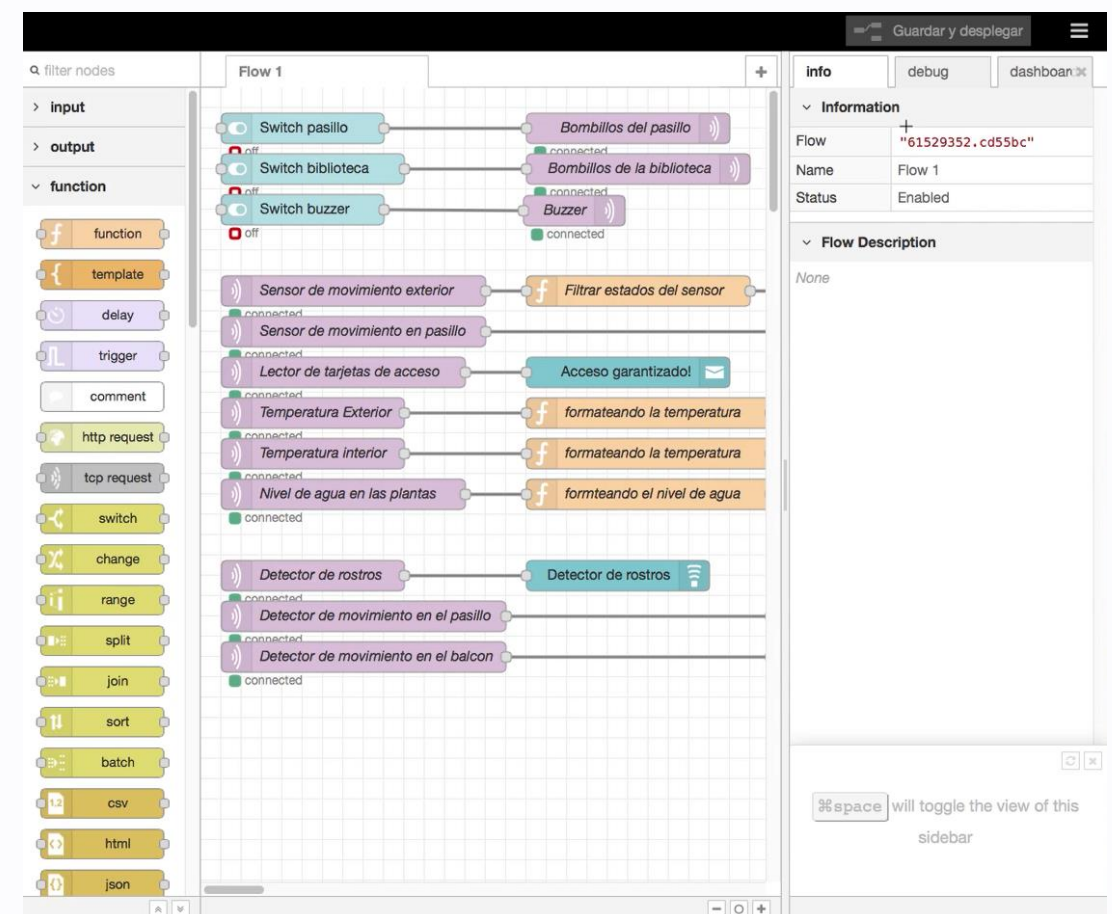
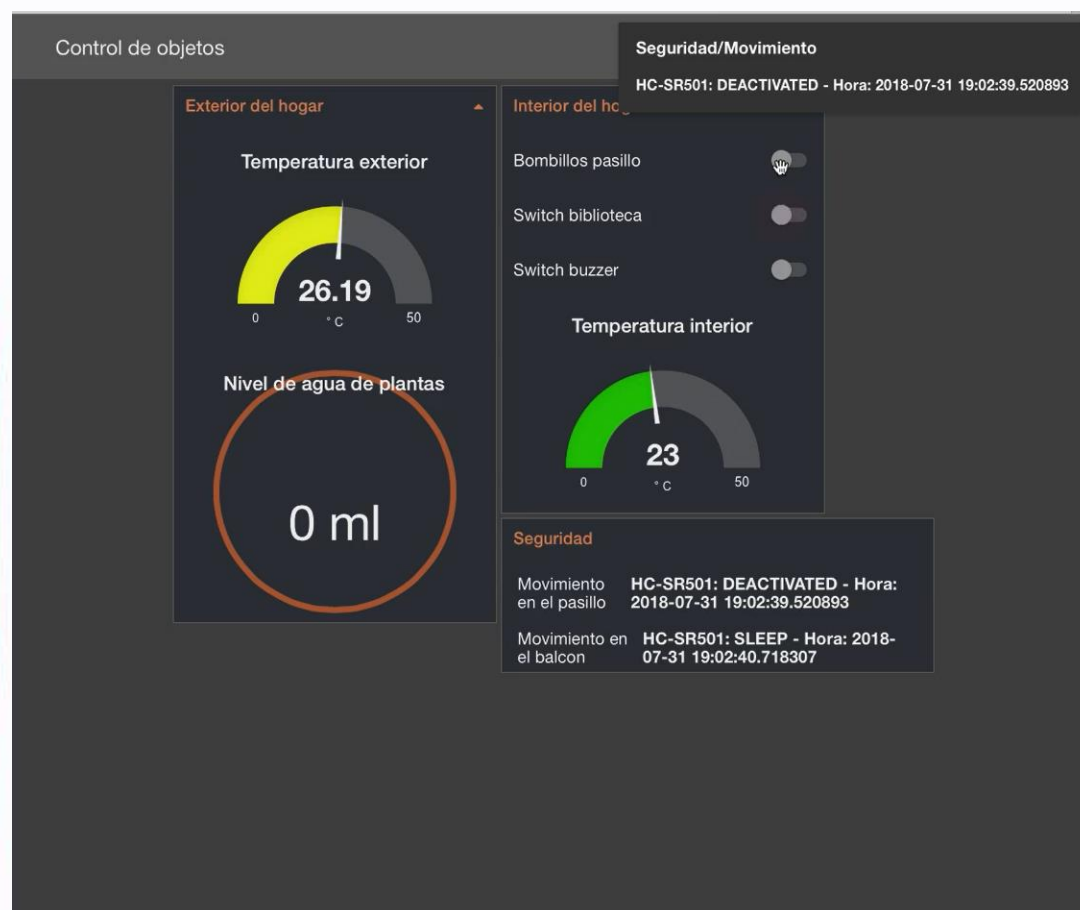
Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos

La aplicación web HAMACA consta de una serie de módulos que se integran sobre una interfaz web que es accesible dentro de una red. Por motivos de seguridad solo los usuarios previamente registrados y autenticados pueden acceder a la información que existe en el sistema.



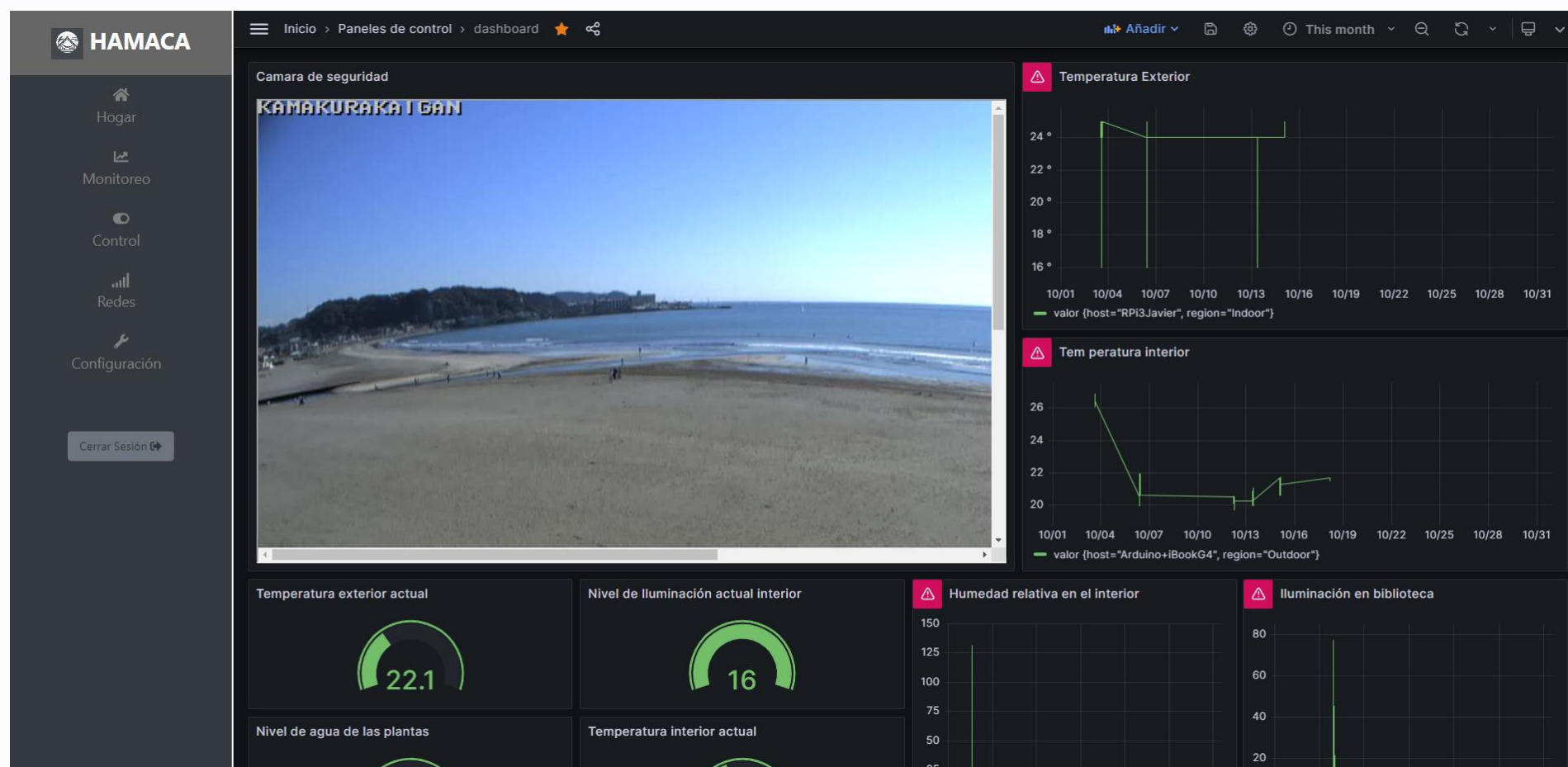
Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos

Los módulos de control y automatización de flujos están embebidos y configurados desde la aplicación Node-Red.



Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos

El modulo de visualización de datos en paneles esta embebido desde la aplicación Grafana.



Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos

En el caso del módulo de Redes se pueden observar las estadísticas y dispositivos conectados al sistema.

 **HAMACA**

Configuración de Redes

 Hogar

 Monitoreo

 Control

 Redes

 Configuración

[Cerrar Sesión](#)

Estadísticas de red

Estadísticas de mensajes MQTT

Mensajes Recibidos: 125538

Mensajes/min: 58

Cantidad de topicos: 4

Topicos activos: 3

Detalles

Topico	Número de Mensajes	Activo	Mensajes/min	Hora	Último Mensaje
Indoor/Temperatura	62738	Activo	28	6 de Octubre de 2019 a las 19:55	DHT11: Temp = 23.0 C - Hora: 2019-10-06 19:55:42.177400
Indoor/Humedad	62791	Activo	28	6 de Octubre de 2019 a las 19:55	DHT11: HR = 40.0% - Hora: 2019-10-06 19:55:42.177542
Outdoor/Temperatura	6	Inactivo	1	4 de Octubre de 2019 a las 14:46	DHT11: Temp = 21.0 C - Hora: 04-10-2019 14:56:45
python/mqtt	3	Activo	1	10 de Octubre de 2023 a las 17:48	mensajes: 1

Página 1 de 1



Herramienta de Automatización, Monitoreo y Análisis de Componentes y Artefactos

En el caso del módulo de Configuración se administran los elementos de las herramientas integradas, así como también la gestión de usuarios.



The screenshot displays the HAMACA web application interface. On the left is a dark sidebar with navigation links: Hogar, Monitoreo, Control, Redes, Configuración, and Cerrar Sesión. The main content area is titled 'Configuración de la Aplicación' and is divided into two sections. The first section, 'Configuración de servicios', contains two entries: 'Ubicación del servicio de Grafana' with the URL 'http://localhost:3000' and 'Ubicación del servicio de Node-RED' with the URL 'http://localhost:1880'. The second section, 'Configuración de Usuarios', includes a 'Crear nuevo usuario' button and a table titled 'Lista de usuarios'. The table has columns for 'Usuario', 'Nombre', 'Apellido', 'Correo', and 'Opciones'. It lists one user: 'pedro' with the email 'pedro.boll22@gmail.com'. The footer of the interface shows 'Pagina 1 de 1'.

Usuario	Nombre	Apellido	Correo	Opciones
pedro			pedro.boll22@gmail.com	 



Demostración



Resultados

Resultados obtenidos de los prototipos

- Se desarrollaron scripts para 18 sensores y actuadores distintos. Durante la operación de los prototipos de dispositivos IoT llevó a la obtención de 2.6 GB de datos no estructurados, 3.1 GB en formato de imágenes y de 1.7 GB en formato de video.
- La elección de las herramientas de visualización, monitoreo y control fueron influenciadas por las características de los datos y la forma de comunicación de los prototipos.
- La captura de información se realizó durante 8 meses.



Resultados

Resultados obtenidos en la aplicación web

- La aplicación web fue construida haciendo uso de la metodología ágil Scrum, en un periodo de 6 sprints de 2 semanas para un total de tiempo de 3 meses de desarrollo, prueba y despliegue continuo en entregables incrementales.
- Se separó la aplicación web del backend de ingesta de los datos provenientes de los dispositivos de forma que solo la base de datos basadas en serie de tiempo fuese el elemento base común a ellos.



Resultados

Resultados obtenidos en la aplicación web

- Se utilizó contenedores exitosamente para poder brindar portabilidad e independencia al sistema.
- Se creó un dashboard donde se podía visualizar el estatus de los 18 sensores y actuadores en los dispositivos, un panel de control para los actuadores así como también y 12 flujos de automatización basados en comportamiento de los dispositivos y sus datos.
- Se gestionaron 2 usuarios para la gestión de la aplicación para validar los roles y privilegios sobre los datos e información dentro del sistema.



Conclusiones



Conclusiones

- Este trabajo de investigación permitió demostrar la factibilidad del desarrollo de una herramienta centralizada de visualización de datos, monitoreo y gestión de procesos concernientes a dispositivos IoT.
- El desarrollo de prototipos de dispositivos IoT permitió identificar las fortalezas y debilidades de los enfoques utilizados para medir y controlar estos artefactos.
- Por el lado del desarrollo de la aplicación web se demostró que es posible integrar en un software que pueda orquestar e integrar herramientas existentes para tareas específicas como visualización de datos, automatización de procesos, monitoreo y control de dispositivos.



Conclusiones

- HAMACA puede ser una alternativa viable para poder centralizar las operaciones inherentes a dispositivos IoT y dentro de un esquema flexible en términos de las herramientas a integrar y módulos presentes.
- Finalmente se puede aprovechar las funcionalidades desarrolladas anteriormente dentro del ámbito académico, investigativo, industrial o uso personal.



Trabajos Futuros



Trabajos Futuros

Trabajos futuros para investigar y desarrollar

Entre los trabajos futuros que este trabajo de investigación abre la oportunidad se sugieren los siguientes:

- Desarrollar nuevas interfaces que permitan y conectarse a dispositivos que utilicen otros protocolos y estándares distintos a MQTT.
- Simplificar y mejorar la interfaz grafica de la aplicación web.
- Desarrollar e integrar nuevas funcionalidades que ayuden a mitigar vulnerabilidades inherentes al uso de un sistema web.
- Nuevos módulos de procesamiento que aprovechen los datos generados.



Referencias

- [1] H. Tschofenig & others. Architectural Considerations in Smart Object Networking. Technical report, Internet Architecture Board, Marzo 2015.
- [2] Sector de Normalización de las Telecomunicaciones de la UIT. Descripción general de Internet de los objetos. Technical report, Union Internacional de Telecomunicaciones, Junio 2012.
- [3] Ari Keranen & Carsten Bormann. Internet of Things: Standards and Guidance from the IETF. <http://www.internetsociety.org/publications/ietf-journal-april-2016/internet-things-standards-and-guidance-ietf>, Abril 2016.



Preguntas





Muchas gracias por su
atención

