

PlantsCare – Documentación Técnica

Alberto Guinda Sevilla

Programa de Especialización en Transformación Digital e IoT

Contenido

| | |
|---|----|
| 1. Resumen ejecutivo..... | 4 |
| 2. Introducción..... | 4 |
| 2.1 Contexto y motivación | 4 |
| 2.2 Objetivos..... | 5 |
| 2.2.1 Objetivo general | 5 |
| 2.2.2 Objetivos específicos | 5 |
| 2.3 Público objetivo | 5 |
| 3. Vigilancia tecnológica y estado del arte | 6 |
| 3.1 Sistemas comerciales analizados..... | 6 |
| 3.2 Empresas referentes en hidroponía | 7 |
| 3.3 Estado actual y tendencias tecnológicas futuras | 7 |
| 4. Análisis previo del proyecto | 8 |
| 4.1 Decisiones técnicas y justificación..... | 8 |
| 4.2 Limitaciones técnicas (alcance, presupuesto, energía)..... | 10 |
| 4.3 Riesgos tecnológicos y medidas preventivas | 11 |
| 4.4 Evaluación básica de durabilidad y consumo energético | 11 |
| 4.5 Flujo completo de datos..... | 12 |
| 4.6 Futuras mejoras previstas | 13 |
| 4.6.1 Conectividad y cobertura | 13 |
| 4.6.2 Inteligencia artificial | 13 |
| 4.6.3 Seguridad | 13 |
| 4.6.4 Energía autosostenible..... | 13 |
| 4.6.5 Monitorización activa y actuadores | 13 |
| 5. Visualización web..... | 14 |
| 5.1 Panel y gráficos en tiempo real | 14 |

| | |
|--|----|
| 5.2 Imagen + predicción IA..... | 14 |
| 6. Bitácora del proyecto..... | 15 |
| 6.1 Cronología basada en el cuaderno de bitácora | 15 |
| 6.2 Cambios técnicos y decisiones | 15 |
| 6.3 Retos encontrados y soluciones aplicadas | 15 |
| 7. Pruebas, evaluación y resultados | 16 |
| 7.1 Validación por módulos | 16 |
| 7.2 Consumo energético y rendimiento..... | 16 |
| 7.3 Precisión de la visión artificial | 16 |
| 7.4 Robustez general del sistema..... | 17 |
| 7.5 Mapeo de la arquitectura sobre el modelo OSI..... | 17 |
| 8. Mejoras futuras y escalabilidad..... | 18 |
| 8.1 Mejoras técnicas identificadas | 18 |
| 8.2 Expansión funcional del sistema | 19 |
| 8.3 Escalabilidad y nuevos contextos de uso..... | 19 |
| 9. Modelo de negocio | 20 |
| 9.1 Análisis DAFO..... | 20 |
| 9.2 Business Model Canvas (BMC) | 20 |
| 9.3 Segmentación de clientes | 21 |
| 9.4 Proyección de ventas | 22 |
| 9.5 Estructura de equipo y gastos | 22 |
| 9.6 ROI y break-even..... | 24 |
| 10. Conclusiones finales..... | 24 |

1. Resumen ejecutivo

PlantsCare es un sistema IoT integral diseñado para monitorizar y automatizar cultivos hidropónicos en ambientes controlados, como invernaderos, cultivos interiores y naves agrícolas. Surge como respuesta a la creciente necesidad de optimizar el trabajo agrícola ante desafíos como la escasez de mano de obra y la falta de relevo generacional en el sector.

Este proyecto está orientado principalmente a pequeños y medianos productores que desean digitalizar sus cultivos sin incurrir en grandes inversiones iniciales. El sistema integra sensores ambientales y acuáticos, cámaras para captura periódica de imágenes, y dispositivos conectados que envían datos en tiempo real a un servidor remoto. Toda esta información es accesible mediante una interfaz web intuitiva, que permite monitorizar el cultivo remotamente y recibir alertas en tiempo real sobre posibles anomalías.

Un aspecto diferencial de PlantsCare es el uso de inteligencia artificial, mediante una red neuronal entrenada especialmente para detectar de forma automática plagas, anomalías o estados de madurez del cultivo, aunque esta versión inicial cuenta aún con potencial para mejorar su precisión debido al limitado conjunto de datos de entrenamiento.

El sistema ha sido validado en entornos reales controlados, demostrando su capacidad funcional en la captura continua de datos y la monitorización remota. Además, PlantsCare propone un modelo de negocio escalable basado en suscripción mensual, facilitando el acceso y adaptación del sistema según la superficie del cultivo.

2. Introducción

2.1 Contexto y motivación

La transformación digital del sector agroalimentario impulsa cada vez más el desarrollo y adopción de sistemas inteligentes y eficientes, especialmente en cultivos hidropónicos y en ambientes controlados. Sin embargo, muchas de estas soluciones tecnológicas suelen ser complejas o demasiado costosas para productores pequeños y medianos. Adicionalmente, el sector enfrenta un preocupante declive en la mano de obra agrícola y escaso relevo generacional.

En este contexto, PlantsCare nace como una solución accesible, realista y efectiva que busca facilitar a los agricultores herramientas sencillas para optimizar procesos, reducir riesgos y automatizar tareas críticas mediante tecnologías como IoT, sensores avanzados, visión e inteligencia artificial.

2.2 Objetivos

2.2.1 Objetivo general

Desarrollar un sistema IoT modular que facilite la monitorización, análisis y soporte a la toma de decisiones en cultivos hidropónicos mediante sensores físicos, visión artificial, procesamiento local y en la nube, accesible a través de una interfaz web sencilla e intuitiva.

2.2.2 Objetivos específicos

- Implementar una red distribuida de sensores para monitoreo ambiental y del agua en tiempo real.
- Capturar y analizar imágenes del cultivo mediante una red neuronal para la detección temprana de plagas, anomalías y estados de madurez.
- Diseñar una arquitectura técnica eficiente que combine procesamiento local con almacenamiento seguro en la nube.
- Crear una plataforma web dinámica para la visualización gráfica de datos, alertas y acceso remoto.
- Realizar validaciones exhaustivas del sistema mediante pruebas funcionales, evaluación del consumo energético y desempeño de la inteligencia artificial.
- Explorar y desarrollar un modelo de negocio sostenible y escalable basado en suscripción mensual.

2.3 Público objetivo

PlantsCare se dirige principalmente a:

- Agricultores pequeños y medianos interesados en digitalizar sus explotaciones agrícolas (invernaderos, sistemas hidropónicos) sin realizar inversiones iniciales elevadas.
- Cooperativas agrícolas, centros educativos o instituciones que desean implementar soluciones IoT en sus programas formativos, proyectos de investigación o iniciativas piloto.

- Proyectos de agricultura innovadora y sostenible, tales como huertos urbanos, agricultura regenerativa y cultivos verticales, que buscan mejorar la eficiencia, trazabilidad y automatización de sus procesos.

3. Vigilancia tecnológica y estado del arte

3.1 Sistemas comerciales analizados

El ecosistema de soluciones agrícolas basadas en IoT y visión artificial ha crecido notablemente en los últimos años, impulsado por la necesidad de optimizar procesos y mejorar la eficiencia del sector agroalimentario. Entre las plataformas comerciales y abiertas analizadas destacan:

Trapview (Eslovenia): Ofrece trampas inteligentes con cámaras integradas y análisis mediante inteligencia artificial para monitoreo en tiempo real de plagas, facilitando decisiones rápidas y reduciendo costos operativos.

Web oficial: <https://trapview.com/es>

Semios (Canadá): Plataforma integral enfocada en monitoreo climático, detección de plagas y riego optimizado para grandes extensiones agrícolas. Combina sensores distribuidos, análisis de datos y predicción de condiciones agrícolas.

Web oficial: <https://semios.com>

Wiseconn (Chile): Especializada en riego inteligente, permite el monitoreo remoto y automatización del riego, optimizando el consumo de agua y mejorando significativamente la salud del cultivo.

Web oficial: <https://wiseconn.com>

Aunque estas soluciones se orientan principalmente a grandes explotaciones, han servido de referencia técnica y visual para PlantsCare, que busca adaptarlas específicamente a las necesidades de pequeños y medianos productores.

3.2 Empresas referentes en hidroponía

Durante la investigación inicial se analizaron experiencias destacadas en hidroponía que validan la viabilidad técnica y comercial del cultivo tecnificado en entornos controlados:

Groots (España): Promueve producción vertical de vegetales en entornos urbanos utilizando sensores avanzados, control automatizado de nutrientes y sistemas cerrados de iluminación y riego.

Web oficial: <https://www.groots.eco>

Video: [Groots - Fábrica de vegetales del futuro](#)

La Yaya Mariana (José Elías, España): Especializada en lechugas hidropónicas cultivadas en interiores, con iluminación artificial y sistemas automatizados de recirculación del agua.

Web oficial: <https://holistic-green.com>

Video: [José Elías - Lechugas hidropónicas](#)

Tomati & Co (España): Proyecto de tomates hidropónicos en interiores, destacando control preciso de temperatura, nutrientes y humedad para obtener productos homogéneos y trazables.

Web oficial: <https://tomatiandco.com>

Video: [Tomates - Hidroponía en naves](#)

Estas iniciativas confirman que los sistemas hidropónicos pueden escalarse de forma efectiva en condiciones controladas y tecnificadas, reforzando así el enfoque estratégico y técnico de PlantsCare.

3.3 Estado actual y tendencias tecnológicas futuras

Durante la fase de diseño del sistema, se evaluaron tecnologías emergentes que, aunque no se integraron inicialmente por limitaciones técnicas y presupuestarias, representan oportunidades importantes para futuras versiones:

- **GSM / LTE / NB-IoT:** Permiten conexión directa y robusta con la nube, ideales para áreas rurales o remotas sin cobertura Wi-Fi fiable.
- **LoRa / LoRaWAN:** Protocolos que ofrecen alcance extendido y bajo consumo energético, ideales para despliegues amplios de sensores agrícolas distribuidos.

Actualmente, PlantsCare opera con redes Wi-Fi usando protocolos como BLE, CoAP, HTTP y MQTT adecuados para ambientes controlados con un alcance estimado de hasta 50 metros. Sin embargo, estas tecnologías futuras podrían ampliar significativamente el potencial de expansión y autonomía del sistema.

4. Análisis previo del proyecto

4.1 Decisiones técnicas y justificación

Se decidió diseñar e implementar una arquitectura modular, escalable y de bajo consumo, con un flujo de datos claro que permitiese desplegar y adaptar el sistema fácilmente según las necesidades de cada cultivo. Las decisiones técnicas buscaron equilibrar fiabilidad, eficiencia energética, costes asequibles y facilidad de mantenimiento:

- **ESP8266:** seleccionado por su bajo consumo (deep-sleep < 0,1 mA, logrado puenteando RST con D0), bajo coste y compatibilidad con CoAP. Captura datos ambientales (DHT22, MQ-135, TSL2561) y acuáticos (DS18B20) cada 30 segundos, generando payloads JSON comprimidos para reducir consumo y ancho de banda. El modo deep-sleep se probó durante el prototipo, pero quedó deshabilitado en la demo para garantizar la continuidad de las lecturas.
- **ESP32-CAM:** recursos limitados obligan a dos modos de operación. Fuera de demanda web, cada 30 segundos captura una foto, la nombra con timestamp (ej. "20250601_121530.jpg"), la guarda en microSD y la envía por HTTP POST al servidor Flask. Al acceder al stream en la web, pasa a transmisión MJPEG continua hasta que el usuario cierra el visor. Para garantizar la estabilidad de la demo no se habilitó light-sleep, pues interrumpe la conexión y podría ocasionar fallos en el envío periódico.
- **Raspberry Pi 4:** actúa como nodo central local (Gateway). Ejecuta tres scripts clave con muestreo cada 30 segundos:

- **tilebox.py**: se conecta por BLE al SensorTile.box Pro, lee la presión atmosférica y publica el valor en MQTT bajo plantscare/tilebox.
- **servidorcoap.py**: expone los endpoints CoAP /aire y /agua, recibe los PUT de los ESP8266 y reemite los datos vía MQTT en plantscare/aire y plantscare/agua.
- **servidorflask.py**: atiende las peticiones HTTP POST de la ESP32-CAM, almacena las imágenes localmente y las reenvía a EC2 para clasificación.

Además corre **Mosquitto** como broker MQTT local. En versiones futuras se añadirá seguridad con TLS en MQTT y DTLS en CoAP.

- **AWS EC2 (Ubuntu Server 22.04 LTS)**: instancia con IP elástica fija que centraliza el procesamiento y almacenamiento de datos:
 - **InfluxDB**: almacena las lecturas sensoriales de aire, agua y presión en series temporales.
 - **SQLite(imagenes.db)**: conserva los últimos resultados de clasificación IA (timestamp, ruta de imagen, etiqueta y probabilidad).
 - **EfficientNet_B0 (.onnx)**: modelo de deep learning preentrenado en ImageNet para clasificación de imágenes. Lo afiné (“fine-tuned”) en Google Colab con un dataset propio de 15 clases de tomates (sanos y con diversas plagas), y lo exporté a ONNX para lograr una inferencia rápida y portátil en la instancia EC2.
 - **mqtt_to_influx.py**: script Python que se suscribe a los topics MQTT de la Raspberry Pi, parsea cada mensaje, convierte el campo ts (epoch) a formato ISO 8601 y escribe los datos sensoriales en InfluxDB.
 - **servidor_modelo.py**: servidor Flask que recibe las imágenes periódicas de la ESP32-CAM vía HTTP POST, carga el modelo ONNX para hacer la inferencia y almacena los resultados de clasificación (timestamp, nombre de fichero, etiqueta y probabilidad) en la base SQLite.

Reglas de entrada configuradas en EC2:

| Puerto | Protocolo | Descripción | Origen |
|--------|-----------|-----------------------------------|-----------|
| 8000 | TCP | Servidor Flask (clasificación IA) | 0.0.0.0/0 |

| | | | |
|------|-----|------------------------------------|-----------|
| 5683 | UDP | CoAP (futuras conexiones directas) | 0.0.0.0/0 |
| 443 | TCP | HTTPS (previsto para futuro uso) | 0.0.0.0/0 |
| 1883 | TCP | MQTT (datos desde Raspberry Pi) | 0.0.0.0/0 |
| 8086 | TCP | InfluxDB (uso interno) | 0.0.0.0/0 |
| 22 | TCP | SSH (gestión servidor EC2) | 0.0.0.0/0 |
| 80 | TCP | HTTP (acceso básico) | 0.0.0.0/0 |

- **Plataforma Web:** construida con Astro, TailwindCSS, React, y TypeScript.

Astro compila y prerenderiza todas las páginas en el servidor, generando HTML estático ultrarrápido, y sólo carga el JavaScript necesario en el cliente donde hay componentes interactivos. Gracias a su arquitectura de “islas” cada componente **React** se hidrata de forma independiente (por ejemplo los gráficos o la sección de clasificaciones), reduciendo el código enviado al navegador y mejorando el rendimiento. **Tailwind**, con su enfoque utility-first, acelera el desarrollo de interfaces limpias y 100 % responsive: defines estilos directamente en la marca HTML con clases semánticas sin escribir hojas de estilo largas. Por último, **TypeScript** aporta tipado fuerte a todo el proyecto, detecta errores en tiempo de compilación y facilita la escalabilidad y el refactoring a medida que el código crece.

Web pública del sistema: <https://plantscareweb.vercel.app>

4.2 Limitaciones técnicas (alcance, presupuesto, energía)

Durante el desarrollo del prototipo surgieron varias restricciones que conviene tener en cuenta:

- **Alcance de red Wi-Fi:** efectivo entre 10 y 50 m en condiciones reales, por lo que el sistema está pensado para espacios cerrados con cobertura estable (invernaderos, interiores).
- **Coste de conectividad:** no se incluyeron tecnologías de largo alcance (GSM/NB-IoT, LoRaWAN) debido al presupuesto disponible.

- **Consumo de Raspberry Pi 4:** entre 2,5 y 3 W en funcionamiento continuo, alto para baterías, pero aceptable en prototipo con alimentación fija.
- **ESP32-CAM:** su hardware no está optimizado para mantener un streaming continuo sin disparar el consumo. Por ello se implementaron dos modos:
 - **Captura periódica:** fuera de demanda web, toma una foto cada 30 s, la guarda en microSD y la envía vía HTTP POST.
 - **Streaming bajo demanda:** si un usuario accede al visor, pasa a MJPEG continuo hasta cerrar la sesión.

El modo light-sleep no se habilitó en la demo para garantizar la fiabilidad de la conexión y evitar pérdidas de envío.
- **Sensores avanzados (pH, EC, NPK):** quedaron fuera del prototipo por falta de disponibilidad, compatibilidad y coste en el plazo de la demo.

4.3 Riesgos tecnológicos y medidas preventivas

| Riesgo | Medida preventiva |
|-----------------------------|--|
| Conexión WiFi inestable | Agrupación local de datos en la Raspberry Pi (MQTT buffer) |
| Desincronización sensores | Uso de identificadores únicos y timestamp sincronizado |
| Pérdida imágenes capturadas | Almacenamiento local en microSD (ESP32-CAM) como respaldo |
| Fallos en clasificación IA | Visualización manual de imágenes en la interfaz web |
| Problemas energía en campo | Modos bajo consumo y evaluación de uso de baterías solares |

4.4 Evaluación básica de durabilidad y consumo energético

- **Sensores (DHT22, MQ-135, TSL2561 y DS18B20):** estabilidad y precisión para prototipos de largo plazo; se recomienda una revisión y limpieza anual de sondas y conexiones.
- **ESP8266:** en deep-sleep su consumo cae a menos de 0,1 mA, lo que permite semanas de funcionamiento con baterías de litio; en transmisión activa ronda los 70 mA.

- **SensorTile.box Pro:** Bluetooth Low Energy (BLE) consume menos de 0,05 mA en reposo y 10–15 mA durante cada lectura de presión. Con muestreo cada 30 s su batería interna dura más de 48 h.
- **ESP32-CAM:** tiene picos de entre 100–200 mA al capturar y transmitir una imagen, pero gracias al muestreo periódico (foto cada 30 segundos) el consumo medio baja por debajo de 20 mA.
- **Raspberry Pi 4:** consumo continuo de 2,5–3 W; es recomendable emplear una fuente de alimentación estable y, en entornos críticos, un sistema de alimentación ininterrumpida (UPS) para evitar reinicios por cortes de energía.
- **AWS EC2:** instancia gestionada en la nube con coste operativo asociado; no se evalúa consumo, ya que se facturan recursos de CPU, almacenamiento y tráfico.

Nota: para la demo hemos configurado todos los nodos con muestreo cada 30 segundos, a fin de generar abundantes datos y demostrar el flujo completo. En un despliegue real conviene adaptar la frecuencia de muestreo según la naturaleza de cada parámetro (por ejemplo, temperatura/humedad cada 1–5 min, CO₂ cada 5–15 min, imágenes cada 15–60 min) para maximizar la vida útil de baterías y componentes sin sacrificar la capacidad de respuesta.

4.5 Flujo completo de datos

- **Dispositivos IoT:**
 - **ESP8266:** envía datos por CoAP a la Raspberry Pi.
 - **ESP32-CAM:** envía imágenes vía HTTP POST.
 - **SensorTile.box Pro:** envía datos BLE leídos por la Raspberry.
- **Raspberry Pi:**
 - Procesa todos los datos y los publica por MQTT en plantscare/aire, plantscare/agua y plantscare/tilebox.
- **EC2:**
 - Recibe los datos y los almacena en InfluxDB y SQLite.
 - Clasifica imágenes con EfficientNet_B0 (.onnx).
- **Interfaz Web:**
 - Actualmente muestra datos simulados. No está conectada directamente a la base de datos Influx (a SQLite sí lo está, mediante la API TypeScript).

4.6 Futuras mejoras previstas

4.6.1 Conectividad y cobertura

Integración de redes de largo alcance: GSM, NB-IoT o LoRaWAN, para funcionar fuera de la cobertura Wi-Fi de invernaderos o naves.

4.6.2 Inteligencia artificial

Ampliación del dataset y aprendizaje colaborativo: a medida que más usuarios empleen PlantsCare y suban imágenes, acumularemos un volumen creciente de fotos etiquetadas. Este flujo continuo permitirá un retraining periódico en la nube, mejorando la precisión del modelo y añadiendo nuevas clases. Con suficiente diversidad de datos, el sistema podrá escalar y adaptarse a otros cultivos más allá del tomate.

4.6.3 Seguridad

- Cifrado TLS para MQTT y DTLS para CoAP.
- Autenticación y autorización con JWT y control de acceso basado en roles.

4.6.4 Energía autosostenible

- Instalación de paneles solares fotovoltaicos.
- Baterías LiFePO₄ con BMS (Battery Management System) para balanceo de celdas, protección contra sobrecarga, sobredescarga y control de temperatura.
- En entornos controlados, posibilidad de usar LiPo de alta densidad energética, siempre con su BMS correspondiente.
- Deep-sleep coordinado en nodos remotos para minimizar consumo cuando no se requieren lecturas ni actuadores.

4.6.5 Monitorización activa y actuadores

- Alertas y notificaciones:
 - Detección de valores fuera de umbral (temperatura, humedad, pH, CO₂, etc.).
 - Envío de alertas push, SMS o correo electrónico al superar límites críticos.
- Actuadores automatizados:
 - Control de riego y válvulas en función de humedad y nutrientes.
 - Ventiladores y humidificadores para ajustar temperatura y humedad.
 - Regulación de intensidad y horario de luz LED para optimizar fotosíntesis.

- Inyección controlada de CO₂ y oxígeno para maximizar el crecimiento.

Con esta hoja de ruta, PlantsCare evolucionará de un sistema de monitorización pasiva a una plataforma de control y respuesta automática, tomando acciones precisas y notificando al usuario únicamente cuando sea necesario.

5. Visualización web

5.1 Panel y gráficos en tiempo real

El sistema PlantsCare incluye un panel de control web interactivo, accesible desde cualquier dispositivo conectado a Internet. Esta interfaz ha sido desarrollada con Astro y React, combinando la eficiencia del renderizado estático con la interactividad dinámica que ofrecen los componentes modernos.

Actualmente los gráficos utilizan datos simulados (arrays estáticos) para validar diseño y responsive. Para la demo ya he integrado con éxito una API en TypeScript que expone las cinco últimas filas de la base SQLite (copia de imagenes.db) y he creado los componentes React/Tailwind que muestran esas cinco clasificaciones reales en pantalla.

Aunque aún no he conectado InfluxDB al frontend, el mismo patrón (consumir un endpoint REST de Influx) se puede replicar sin dificultad, puesto que el script `mqtt_to_influx.py` ya vuelca los datos en InfluxDB y basta exponer un endpoint similar para integrarlo en los gráficos.

El diseño de la interfaz es responsive y permite:

- Visualizar valores simulados de temperatura, humedad, CO₂, luminosidad y temperatura del agua.
- Alternar entre vistas diaria, semanal, mensual o anual usando la librería de gráficos Recharts.
- Consultar los datos en una presentación clara y adaptada a cualquier dispositivo.

5.2 Imagen + predicción IA

El panel ya consume la API TypeScript `/api/clasificaciones`, que devuelve las cinco últimas entradas de imagenes.db copiada desde EC2 al proyecto. Los componentes React/Tailwind que he implementado muestran en tarjetas la foto (ruta `public/imagenes`), el timestamp, la etiqueta

y la probabilidad real de cada clasificación.

En próximas iteraciones bastará con sustituir esas rutas estáticas por las cámaras en vivo y enlazar el mismo endpoint para que las imágenes y predicciones fluyan en tiempo real desde el backend.

6. Bitácora del proyecto

6.1 Cronología basada en el cuaderno de bitácora

Durante el desarrollo de PlantsCare se ha llevado un registro detallado de avances técnicos, decisiones, cambios, pruebas y problemas encontrados. Esta documentación ha sido recogida en un archivo PDF elaborado durante los últimos 9 meses.

- Fechas y descripción de hitos clave del proyecto.
- Cambios de hardware o software motivados por pruebas o fallos.
- Ideas descartadas y sus justificaciones.
- Exploración de tecnologías nuevas, incluso si no llegaron a integrarse.
- Problemas técnicos relevantes y sus soluciones.

6.2 Cambios técnicos y decisiones

Algunas decisiones clave modificadas sobre la marcha fueron:

- Sustitución de Amazon S3 por almacenamiento local en EC2.
- Reducción del número de sensores en el prototipo para optimizar recursos.
- Separación de funciones entre dispositivos (streaming y captura de imágenes).
- Rediseño de la interfaz web y simplificación tras pruebas de usuario.

6.3 Retos encontrados y soluciones aplicadas

- Problemas de memoria en ESP32-CAM al intentar streaming y captura periódica simultánea: se mantuvieron funcionalidades por separado, priorizando la captura automática.
- Inestabilidad de conexiones WiFi: se implementó reconexión automática y desactivación de servicios innecesarios.
- Fallos en la sincronización horaria: se añadió reinicio programado y ajuste horario vía servidor.

- Latencia en web: se optimizó con Astro, carga diferida y división de componentes.

7. Pruebas, evaluación y resultados

7.1 Validación por módulos

Cada componente fue probado de forma autónoma antes de su integración:

- **ESP8266:** lecturas estables, entrada/salida de deep sleep y comunicación CoAP correcta.
- **ESP32-CAM:** captura puntual, almacenamiento en SD, envío HTTP POST funcional.
- **SensorTile.box Pro:** conexión BLE correcta y lecturas válidas cada 30 segundos.
- **Raspberry Pi:** recepción, reenvío MQTT y servidor Flask operativo.
- **EC2:** recepción sin pérdida, InfluxDB funcional, IA integrada.
- **IA:** clasificaciones ONNX funcionales en condiciones controladas.

7.2 Consumo energético y rendimiento

- **ESP8266:** <0,1 mA en reposo, 3 s activos por ciclo.
- **ESP32-CAM:** 180–200 mA en picos, light sleep entre eventos.
- **SensorTile.box Pro:** bajo consumo, ya tiene batería.
- **Raspberry Pi:** consumo estable ~2.5–3 W/h, justificable.

7.3 Precisión de la visión artificial

- El modelo de clasificación IA se incluyó como prueba de concepto, dado que el enfoque principal del curso es IoT y gestión de datos.
- Datasets limitados y de baja calidad, descargados de <https://universe.roboflow.com/search?q=tomates>

mezclando varias imágenes clasificadas en 15 categorías distintas: 'acaros_arana', 'danado', 'maduro', 'mancha_bacteriana', 'mancha_foliar_de_septoria', 'mancha_ocular', 'mildiu_polvoriento', 'moho', 'pasado', 'sanos', 'tizon_tardio', 'tizon_temprano', 'verde', 'virus_del_mosaico', 'virus_del_rizado_amarillo'.

A pesar de estas clases, la mayoría de predicciones actuales devuelven "maduro" como salida, lo que refleja una precisión muy baja y sobreajuste claro.

El modelo fue entrenado en Google Colab con EfficientNet_B0 y exportado a ONNX, pero su rendimiento es insuficiente para aplicaciones reales.

Se considera un primer intento funcional, con margen de mejora importante. La implementación demuestra el flujo completo de captura, envío, inferencia y almacenamiento, aunque los resultados no son válidos para toma de decisiones productivas todavía.

7.4 Robustez general del sistema

- **Tolerancia a fallos:** cada componente está diseñado para reconectar automáticamente ante caídas de red o cortes de energía, garantizando continuidad operativa.
- **Separación de responsabilidades:** los dispositivos están desacoplados funcionalmente (sensores, cámara, procesamiento, almacenamiento), lo que reduce el impacto de fallos aislados.
- **Recuperación tras errores:** el sistema reinicia automáticamente sus servicios críticos tras fallos, sin requerir intervención manual.
- **Modularidad efectiva:** permite sustituir, añadir o actualizar módulos (hardware o software) sin afectar al conjunto, facilitando el mantenimiento y la escalabilidad.

El sistema ha sido validado con éxito en sesiones continuas de varios días, y está listo para ser evaluado en entornos reales, tanto en contextos educativos como en pilotos agrícolas experimentales.

Aunque aún se encuentra en fase de prototipo, PlantsCare demuestra que es posible construir soluciones IoT agrícolas modulares, resilientes y con visión de futuro, listas para evolucionar hacia sistemas autónomos de agricultura inteligente.

7.5 Mapeo de la arquitectura sobre el modelo OSI

Aunque PlantsCare se ha diseñado priorizando eficiencia y sencillez, toda la arquitectura puede mapearse de forma directa sobre el modelo OSI de comunicaciones, lo que facilita la comprensión y futuras integraciones estándar. El desglose es el siguiente:

- **Capa física:** Placas ESP8266, ESP32-CAM, SensorTile.box Pro, Raspberry Pi, cableado, sensores y actuadores conectados.

- **Capa de enlace:** Wi-Fi (IEEE 802.11) y Bluetooth Low Energy (BLE) para comunicación local sin cables.
- **Capa de red:** Uso de direccionamiento IP (tanto privado en LAN local como público en EC2 de AWS con IP elástica).
- **Capa de transporte:** Protocolo UDP para CoAP (transmisión eficiente de datos sensoriales) y TCP para HTTP/MQTT (fiabilidad y persistencia en comunicaciones críticas).
- **Capa de sesión:** Gestión de sesiones MQTT y mantenimiento de conexiones HTTP persistentes cuando es necesario.
- **Capa de presentación:** Codificación y compresión de datos en formato JSON (sensores), JPEG (imágenes), almacenamiento en bases de datos estructuradas (InfluxDB y SQLite) y ONNX como formato de serialización del modelo de IA.
- **Capa de aplicación:** Servicios y scripts como el backend Flask, broker MQTT, servidor CoAP, clasificación IA y la interfaz web React.

8. Mejoras futuras y escalabilidad

8.1 Mejoras técnicas identificadas

Aunque PlantsCare ya funciona como prototipo completo, existen mejoras claras que permitirían aumentar su eficiencia, robustez y adaptabilidad sin rediseñar la arquitectura:

- **Interfaz de configuración remota:** para modificar parámetros como intervalos de captura o límites de alerta sin necesidad de reprogramar físicamente los nodos.
- **Transmisión en lote:** que permita acumular lecturas antes de enviarlas, reduciendo la frecuencia de conexión y el consumo energético.
- **Unificación de sensores y cámara:** usando un microcontrolador más potente que gestione todo desde un único punto, siempre que se garantice estabilidad entre RAM, CPU y consumo.
- **Optimización del clasificador IA:** mediante datasets más amplios y reales, idealmente aportados por los propios usuarios, junto con técnicas de aprendizaje incremental.

- **Gestión avanzada de errores y red:** reintentos automáticos, colas de envío local, y almacenamiento temporal en caso de pérdida de conectividad.

Estas mejoras representan el siguiente paso lógico para escalar PlantsCare a contextos más exigentes.

8.2 Expansión funcional del sistema

Una vez consolidada la fase de monitorización inteligente, PlantsCare puede evolucionar hacia un sistema autónomo capaz de intervenir activamente en el cultivo:

- **Automatización del riego:** control de válvulas en función de parámetros como humedad, pH o temperatura del agua.
- **Dosificación de nutrientes:** gestión activa del abonado hidropónico según las etapas de crecimiento y niveles de EC/NPK.
- **Respuestas autónomas ante eventos:** por ejemplo, activar ventilación si sube la temperatura o encender iluminación en caso de luz insuficiente.

Este tipo de funciones permitirían cerrar el bucle entre medición, análisis y acción, acercando PlantsCare al concepto de agricultura de precisión totalmente automatizada.

8.3 Escalabilidad y nuevos contextos de uso

El diseño modular y flexible del sistema facilita su adaptación a distintos cultivos, entornos y objetivos. Algunas líneas de aplicación previstas son:

- Cultivos de hoja como lechuga, espinaca o albahaca, donde el control hídrico y lumínico es más crítico que la imagen.
- Invernaderos verticales o urbanos, donde el espacio reducido exige automatización avanzada y mínima intervención humana.
- Entornos educativos y de investigación, como herramienta pedagógica para enseñar conceptos de IoT, sostenibilidad, programación o análisis de datos.

Además, el sistema puede adaptarse fácilmente a formatos domésticos, como macetas inteligentes o cultivos ornamentales, simplificando los componentes, pero manteniendo la filosofía de sensorización conectada.

9. Modelo de negocio

9.1 Análisis DAFO

El análisis DAFO realizado sobre PlantsCare identifica las siguientes claves:

Fortalezas:

- Desarrollo propio e integral de hardware, software y modelo de IA.
- Modularidad y adaptabilidad para distintos cultivos y tamaños.
- Sistema completo: sensores de aire, agua y visión artificial, con interfaz web moderna.
- Escalabilidad real gracias al uso de servicios cloud (EC2).

Debilidades:

- Falta de pruebas extensivas en condiciones agrícolas reales.
- Necesidad de soporte técnico inicial en la instalación.
- Carencia de carcasas industriales para todos los sensores.
- Dependencia de varios protocolos, lo que exige un cierto nivel técnico.

Oportunidades:

- Expansión a otros cultivos (lechuga, fresa, pepino, etc.).
- Digitalización acelerada del sector agrícola por exigencias normativas y fondos europeos.
- Futuro control automatizado de riego y nutrientes basado en datos reales.
- Acuerdos con fabricantes internacionales para reducir costes y escalar la producción.

Amenazas:

- Competencia creciente en el sector agrotech.
- Limitaciones de conectividad en entornos rurales.
- Posible resistencia cultural al uso de tecnología por parte de agricultores tradicionales.
- Cambios regulatorios en agricultura digital o importación de hardware.

9.2 Business Model Canvas (BMC)

PlantsCare se estructura sobre un modelo B2B y B2C, destacando:

Propuesta de valor:

Plataforma integral y modular de IoT agrícola, adaptada a cualquier tamaño de explotación y capaz de detectar plagas de forma temprana, optimizar el uso de agua y nutrientes, y ofrecer trazabilidad en tiempo real.

Segmentos de clientes:

- Productores agrícolas de valor (hortalizas especiales, floricultura).
- Agricultores independientes en hidropónico/invernadero.
- Empresas agrícolas medianas y grandes que buscan eficiencia y trazabilidad.
- Cooperativas y asociaciones agrarias.
- Centros educativos, investigación y proyectos urbanos o sostenibles.
- Canales y relación con el cliente.
- Web con demo y contacto directo.
- Distribuidores especializados y presencia en ferias.
- Red de técnicos freelance para instalación local.
- Soporte remoto, actualizaciones OTA y mantenimiento incluido.

Estructura de costes:

- Fabricación y entrega de kits modulares.
- Infraestructura cloud (EC2).
- Equipo de desarrollo y soporte técnico.
- Gastos de logística y marketing.
- Retribución a instaladores y técnicos remotos.

Fuentes de ingresos:

- Suscripción mensual escalable por kit activo y sensores instalados.
- Ampliaciones: sensores adicionales, módulos IA, funciones premium.
- Tarifa inicial/fianza por dispositivos, con penalización por no devolución.
- Ingresos por instalación y soporte avanzado.

9.3 Segmentación de clientes

PlantsCare está pensado para distintos perfiles:

- Empresas agrícolas medianas y grandes que requieren control preciso y reducción de riesgos.
- Agricultores particulares que buscan una solución asequible y plug & play para cultivos hidropónicos o invernaderos domésticos.
- Cooperativas interesadas en digitalizar explotaciones y gestionar varias instalaciones desde un mismo panel.
- Centros educativos/investigación para proyectos formativos y monitorización experimental.
- Agricultura urbana y sostenible: colectivos, startups y proyectos enfocados a la eficiencia hídrica y medioambiental.

La flexibilidad en la suscripción mensual y su modularidad eliminan barreras de entrada, haciendo PlantsCare accesible tanto a usuarios individuales como a organizaciones.

9.4 Proyección de ventas

Dos configuraciones de producto:

- **Producto A:** kit básico para superficies hasta 10 m², con sensores de aire, agua y cámara.
- **Producto B:** kit ampliado para superficies de hasta 50 m².

Precios estimados: 24,95 € al mes (Producto A) y 99 € al mes (Producto B).

Estrategia comercial:

- Validación técnica inicial.
- Presencia en ferias y canales agrícolas.
- Marketing digital y recomendaciones entre usuarios.
- Mejora continua del producto y soporte técnico.

Las previsiones, reflejadas en el Excel adjunto, anticipan un crecimiento progresivo y sostenido, con ingresos recurrentes que permiten escalar sin depender de financiación externa.

9.5 Estructura de equipo y gastos

La estructura de equipo prevista para PlantsCare está optimizada para escalar el negocio y cubrir todas las áreas clave desde el inicio:

Equipo mínimo:

- **CEO/Coordinador general:** Responsable de la gestión global y estrategia.
- **3 desarrolladores remotos (frontend y backend),** seleccionados por eficiencia y disponibilidad.
- **2 especialistas en IA/visión artificial:** Mejoran continuamente el modelo de clasificación de imágenes.
- **1 comercial:** Encargado de captación de clientes y relaciones comerciales.
- **1 agente de soporte:** Responde a incidencias y dudas técnicas.
- **Instaladores freelance:** Contratados por montaje y configuración de sistemas en cliente.
- **Call center subcontratado:** Para atención y soporte automatizado, con escalabilidad según demanda.

Gastos principales contemplados:

- Producción de kits (hardware y sensores).
- Salarios del equipo técnico, comercial y soporte.
- Infraestructura cloud (AWS EC2 y servicios asociados).
- Campañas publicitarias y marketing digital.
- Logística e instalación.
- Call center y atención al cliente (subcontratación).
- Compra de portátiles para el equipo técnico.
- Inversión en I+D para innovación y mejora continua.
- Imprevistos y contingencias para cubrir cualquier necesidad operativa.

Esta combinación de personal propio y recursos subcontratados permite adaptarse rápidamente al crecimiento, controlar costes y mantener un servicio de calidad.

9.6 ROI y break-even

El análisis financiero muestra que, con la estructura de costes y el ritmo de captación previsto, el **punto de equilibrio (break-even) se alcanza en el mes 20.**

El retorno de la inversión (ROI) se cumple aproximadamente en el mes 33, según los cálculos reflejados en el Excel adjunto.

Este periodo refleja el esfuerzo inicial necesario para lanzar una solución tecnológica robusta, invertir en equipo, cloud y marketing, y consolidar una base de clientes suficiente para cubrir los costes fijos y variables.

La estructura de costes ha sido diseñada para no crecer de forma proporcional al número de clientes, de modo que el margen operativo mejora progresivamente con la escala y permite reinvertir en innovación, soporte y expansión.

10. Conclusiones finales

PlantsCare es mucho más que un prototipo técnico: representa una visión realista y escalable para llevar el IoT y la inteligencia artificial al sector agrícola, especialmente en cultivos controlados, invernaderos y espacios urbanos.

Entre los logros clave destacan:

- Implementación funcional de un sistema IoT distribuido, eficiente y de bajo consumo.
- Transmisión segura y modular mediante protocolos adaptados al entorno (CoAP, MQTT, HTTP, BLE).
- Infraestructura flexible y escalable en cloud, preparada para evolucionar según las necesidades.
- Interfaz web profesional y moderna, orientada a la visualización y el control remoto.
- Integración de IA para clasificación visual, aunque aún en fase experimental.
- Modelo de suscripción que facilita el acceso a esta tecnología tanto a aficionados como a profesionales.

PlantsCare permite digitalizar el campo, reducir costes, optimizar recursos y anticiparse a problemas, facilitando una agricultura de precisión para todos los niveles, desde particulares hasta grandes explotaciones.

A pesar de las limitaciones del prototipo, la experiencia adquirida es aplicable a otros cultivos y mercados. El sistema está listo para pruebas externas, y su arquitectura modular facilita tanto la evolución técnica como la adaptación comercial.