

Grado en Ingeniería de Tecnologías de Telecomunicación
2018-2019

Trabajo Fin de Grado

Diseño de implementación de un sistema para monitorización de cultivos y equipos agrícolas

Clemente Rodríguez Arráez

Tutora

Florina Almenares Mendoza

Leganés, 14 de octubre de 2019



Esta obra se encuentra sujeta a la licencia Creative Commons
Reconocimiento – No Comercial – Sin Obra Derivada

TRABAJO DE FIN DE GRADO

DISEÑO DE IMPLEMENTACIÓN DE UN SISTEMA PARA MONITORIZACIÓN DE CULTIVOS Y EQUIPOS AGRÍCOLAS

AUTOR: CLEMENTE RODRÍGUEZ ARRÁEZ

TUTORA: FLORINA ALMENARES MENDOZA

EL TRIBUNAL

PRESIDENTE: ANTONIO ARTÉS RODRÍGUEZ

SECRETARIO: CARLOS SEVILLA SALCEDO

VOCAL: JOSÉ MANUEL SÁNCHEZ PEÑA

SUPLENTE: ANTONIO LÁZARO BLANCO

Realizado el acto de defensa y lectura del Trabajo Fin de Grado el día 14 de octubre de 2019 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la siguiente CALIFICACIÓN:

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

En primer lugar, a mi tutora, Florina Almenares, por guiarme en el TFG hasta el final con unas ideas fabulosas para mejorarlo. Siempre ahí, ¡muchas gracias!

A toda mi familia por mostrarme todo su apoyo y comprensión que han tenido conmigo todo este tiempo.

A todos mis amigos de Corral de Almaguer: Dani, Jorge, Eugenio, Xermán, Alberto, Alex, Miguel y a todos los demás.

También, a mis compadres de *Los ajos las Pedroñeras*: Miguel, Diego, Pablo, Carlos, Caste y por supuesto, Joseda y María. Vuestro apoyo siempre ha estado ahí, ¡muchísimas gracias!

Además, mis amigos de la Universidad, *Los Telecomos con Força*: Josué González, Javier Yuste, Javier Albert, David Ventura, Fran Hong, José Luis Arias y Rafa Osuna. A todos vosotros y a muchos más por hacer de la carrera, un momento muy divertido en mi vida.

También... I'd like to thank all my friends that I met in Maryland (USA), especially Matt Flaherty, Juan Abascal and Andrés Carrillo for helping me with many concepts for this project. Without your help, I would not finish this capstone... Thank you a lot ;).

Finalmente, a Pedro Antonio García-Gasco Real por contagiarme su gran motivación en el ámbito de las ciencias, y en especial de la robótica, además de enseñarnos que nuestro pueblo es un tesoro.

...Y por supuesto a ti, ¡gracias!

Resumen

El fin de este proyecto es el diseño de un sistema utilizando la tecnología actual que haga uso de dos conceptos actuales: Internet de las cosas y redes de sensores. Se ha decidido enfocar este desarrollo en la extracción de datos medioambientales. Particularmente, en el diseño de implementación de un sistema para monitorización de cultivos (en especial de la vid) y equipos agrícolas.

Los objetivos principales de este proyecto son la mejora del riego de las plantas, automatizando esta tarea para optimizar el gasto de agua que se extrae de los acuíferos, y la seguridad de las máquinas agrícolas valiosas tales como los motores de gasoil que generalmente se encuentran desprotegidos en el exterior.

Las tareas principales del sistema son la recopilación de los datos de los sensores, el análisis de estos, el registro de todas las actividades que realice el sistema y el envío de alertas al usuario. Los datos se podrán acceder de tres maneras: en una página web, recibiendo las alertas vía SMS y correo electrónico y utilizando una conexión directa con el sistema utilizando la terminal.

El sistema se compone de varios elementos principales: un coordinador que realice todas las tareas instalado en una Raspberry Pi, dos aplicaciones destinadas al riego y a la seguridad, un sensor de humedad de suelo, un sensor de movimiento, un giroscopio y, además, se ha utilizado un módem que envía los datos en cualquier lugar del mundo utilizando la tecnología GSM.

Palabras clave: internet de las cosas, redes de sensores, GSM, monitorización, análisis de datos, seguridad agrícola, riego automático, Hologram.

Abstract

The goal of this project is to design a system by using the current technology in order to apply the concept of Internet of things and Wireless Sensor Networks. The extraction of environmental data is the main point of this development, especially, a design of a system implementation for farming (vineyards in particular) and agricultural equipment monitoring.

The main objectives in this capstone project are the plants' irrigation improvement, by making this task automatic to optimize the use of water from aquifers, and the agricultural valuable machines security improvement such as diesel engines that are unprotected in farmers' lands.

The main tasks of the system are the collection of data from the sensors, their analysis, the recording of all the activities carried out by the system and the sending of alerts to the user. The data can be accessed in three ways: on a web page, receiving alerts via SMS and email and using a direct connection to the system by using the terminal.

The system consists of several elements: a coordinator who performs all the tasks installed in a Raspberry Pi, two applications for irrigation and safety, a soil moisture sensor, a motion sensor, a gyroscope and, in addition, a modem that sends data from anywhere in the world by using GSM technology.

Keywords: Internet of Things, Wireless Sensor Networks, GSM, monitoring, data analysis, agricultural security, automatic irrigation, Hologram.

Índice general

1. INTRODUCCIÓN Y OBJETIVOS	1
1.1. Contexto	1
1.2. Motivación	2
1.3. Objetivos	4
1.3.1. Objetivos específicos	4
1.4. Medios utilizados	5
1.4.1. Software	5
1.4.2. Hardware	6
1.5. Entorno Socioeconómico	11
1.6. Estructura de la memoria	12
2. ESTADO DEL ARTE	14
2.1. Internet de las Cosas	14
2.1.1. Evolución de las tecnologías con el Internet de las Cosas	17
2.1.2. Aplicaciones del Internet de las Cosas	18
2.2. Red de sensores inalámbricas	22
2.2.1. Usabilidad de las Redes de sensores inalámbricas	22
2.2.2. Arquitectura de la red de sensores en la agricultura	23
2.2.3. Arquitectura de los nodos sensores	25
2.3. Tecnología y estándares usados en la agricultura	25
2.3.1. Zigbee	25
2.3.2. WiFi	27
2.3.3. Redes móviles y su evolución	27
2.4. Tecnologías utilizadas	28
2.4.1. Raspberry Pi	28

2.4.2. Sensores	29
2.4.3. Raspbian OS.	29
2.4.4. Python	30
2.4.5. Hologram.	30
2.4.6. ThingSpeak	31
2.4.7. Google Maps	31
2.4.8. MobaXterm	31
2.4.9. R y RStudio	32
2.4.10. Draw.io.	32
2.4.11. Comunicación	32
2.4.12. API REST	33
3. ANÁLISIS Y DISEÑO	34
3.1. Requisitos del sistema	34
3.1.1. Requisitos funcionales	35
3.1.2. Requisitos no funcionales.	39
3.1.3. Casos de uso	41
3.1.3.1. Actores.	42
3.1.3.2. Diagramas de los casos de uso.	42
3.1.3.3. Descripción de los casos de uso	43
3.2. Arquitectura del sistema	50
4. IMPLEMENTACIÓN	52
4.1. Contexto.	52
4.2. Programa de riego – riego.py.	53
4.2.1. Análisis de la humedad	54
4.2.2. Cálculo del agua alcanzada	54
4.2.3. Cálculo de agua diaria disponible	55
4.2.4. Envío datos a ThingSpeak	55
4.2.5. Creación del fichero “log_riego.txt”	57
4.3. Detección de robo	58
4.3.1. Sistema de activación y desactivación del programa	59
4.3.2. Detección de cuerpos térmicos en movimiento	60
4.3.3. Análisis del movimiento angular y lineal del motor	61
4.3.4. Localización del dispositivo	64

4.3.5. Envío de alertas al usuario	65
4.3.6. Inicio de parámetros y librerías.....	67
4.3.7. Creación de archivos “log.txt” y “alertas.txt”.....	68
4.4. Terminal	68
4.4.1. Comandos	71
4.5. La unión de todas las piezas	72
4.6. Simulación	73
5. PRUEBAS	78
6. PLANIFICACIÓN Y PRESUPUESTO	81
6.1. Planificación	81
6.2. Presupuesto.....	84
6.2.1. Coste de materiales	84
6.2.2. Otros costes	85
6.2.3. Coste de personal	86
6.2.4. Coste total del proyecto.....	86
7. MARCO REGULADOR	87
7.1. Normas y legislación	87
7.2. Licencias y tecnologías.....	88
7.3. Propiedad intelectual.....	89
8. CONCLUSIONES Y LÍNEAS FUTURAS	90
8.1. Conclusiones.....	90
8.2. Líneas futuras	91
REFERENCIAS	93
ANEXO 1. SUMMARY	106
1.1. Introduction	106
1.2. Tools and devices	107
1.2.1. Software	107
1.2.2. Hardware.....	107
1.3. Objectives.....	109
1.4. Implementation	110
1.4.1. Riego.py	110
1.4.1.1. Humidity analysis	110

1.4.1.2. Sending data to ThingSpeak	111
1.4.1.3. Log_riego.txt	111
1.4.2. Seguridad.py	112
1.4.2.1. System activation and deactivation	112
1.4.2.2. PIR sensor.	112
1.4.2.3. Gyroscope.	113
1.4.2.4. Device location	113
1.4.2.5. Sending alerts to the user	113
1.4.2.6. Log_seguridad.txt and alertas.txt	113
1.4.3. Terminal	114
1.4.4. Data consumption	114
1.4.5. All in one	114
1.5. Testing	115
1.6. Conclusion	115
ANEXO 2. AMPLIACIÓN SOBRE LOS CONCEPTOS SOBRE LA VID	116
2.1. El ciclo vegetativo de la vid	117
2.1.1. Brotación	117
2.1.2. Foliación.	117
2.1.3. Floración.	117
2.1.4. Fecundación y fructificación.	117
2.1.5. Envero y maduración.	118
2.1.6. Vendimia y parada.	118
2.2. Necesidad hidrológica	118
2.3. Índice de Precipitación	118
2.3.1. Evapotranspiración	119
2.4. Precipitación en la temporada 2016 – 2017.	123
2.5. Una solución a corto plazo	126
ANEXO 3. ANÁLISIS DEL CONSUMO DE DATOS EN LA NUBE	128
3.1. Informes de uso	128
3.2. Uso en directo	129
ANEXO 4. TABLA DE RADIACIÓN SOLAR EXTRATERRESTRE	130
ANEXO 5. GLOSARIO DE ACRÓNIMOS Y DEFINICIONES	131

Índice de figuras

1.1. Ejemplo de motor de gasoil – DEUTZ 33 kVA	3
1.2. Ejemplo de panel solar destinado al riego	3
1.3. Raspberry Pi 3 Model B	6
1.4. Higrómetro FC-28	6
1.5. Vista con detalle de la placa de medición estándar	7
1.6. Ejemplo del ADC MCP3008	7
1.7. Esquema de Raspberry Pi, el MCP3008 y el FC-28	8
1.8. Sensor PIR HC-SR501	8
1.9. Esquema de detección del sensor HC-SR501	9
1.10. Esquema de detección de movimiento	9
1.11. Vista general del giroscopio y acelerómetro MPU6050	10
1.12. Ejemplo tarjeta SIM Hologram	10
1.13. Módem - Hologram Nova	11
1.14. Comando de instalación.	11
2.1. Internet de las Cosas	15
2.2. Número de dispositivos conectados a Internet en el mundo	16
2.3. Número de dispositivos conectados usados por persona en España	16
2.4. Aplicaciones del IoT en la agricultura.	18
2.5. Uso del dron para monitorización de una viña en Argentina	19
2.6. Mejora de la domótica con el IoT	20
2.7. Ejemplo del uso de sensores para conducción autónoma	20
2.8. Internet de las Cosas Médicas	21
2.9. Internet Industrial de las Cosas	21

2.10. Ejemplo de red de sensores inalámbrica.	22
2.11. Ejemplo de módulo de red Zigbee	26
2.12. Red de Sensores Zigbee	27
2.13. Ejemplo de triangulación por GSM para localizar un dispositivo	28
3.1. Esquema del escenario 1 – Interfaz Web	42
3.2. Esquema del escenario 2 - Terminal.	43
3.3. Esquema del escenario 3 – Servidor.	43
3.4. Arquitectura ideal del sistema.	50
4.1. Diagrama de flujo – riego.py.	53
4.2. Tiempo motor (activo) y tiempo parada cada hora.	55
4.3. Configuración de los parámetros para acceder a ThingSpeak	56
4.4. Función para enviar mensaje a la plataforma	56
4.5. Configuración del canal en la página web	56
4.6. Gráfica de la humedad en tiempo real	57
4.7. Ejemplo de “log_riego.txt”	57
4.8. Esquema de funcionamiento – seguridad.py	58
4.9. Sentencia condicional con el sensor PIR	61
4.10. Pruebas del sensor MPU6050 (Aceleración y Rotación).	61
4.11. Posición del sensor MPU6050	62
4.12. Vista de la aceleración en los diferentes ejes y el MPU6050.	62
4.13. Diferentes posiciones del giroscopio	63
4.14. Sentencia condicional con sensor MPU6050	63
4.15. Raspberry Pi conectado al módem Hologram Nova	64
4.16. Sentencia condicional para enviar la localización	64
4.17. Envío de mensaje con la etiqueta PIR_SENSOR, GYRO_SENSOR y Location	65
4.18. Página de rutas en Hologram Cloud	66
4.19. Ejemplo de configuración de ruta SMS	66
4.20. Ejemplo de configuración de ruta de correo electrónico.	67
4.21. Ejemplo log_seguridad.txt	68
4.22. Ejemplo alertas.txt.	68

4.23. Activación de Spacebridge en la SIM seleccionada	69
4.24. Inicio del programa Spacebridge.	69
4.25. Selección de SIM y puertos	70
4.26. Túnel creado.	70
4.27. Acceso por SSH	71
4.28. Sistema de monitorización y el motor de gasoil	73
4.29. Sistema encima del motor de gasoil	74
4.30. Motor de gasoil con el sensor PIR en posición para detectar movimiento.	74
4.31. Bienvenida al programa y activación y desactivación del sistema de seguridad	75
4.32. Alertas vía SMS	76
4.33. Alerta y envío de localización.	76
4.34. Notificaciones por correo electrónico – sensor PIR	77
4.35. Notificaciones por correo electrónico – giroscopio.	77
4.36. Notificaciones por correo electrónico – localización.	77
6.1. Diagrama de Gantt	83
A.1.1. Architecture of the system	110
A.2.1. Cepas en vaso	116
A.2.2. Cepas en espaldera	117
A.2.3. Temperaturas máximas y mínimas en la campaña 2016 – 2017	121
A.2.4. Radiación solar incidente (Fórmula de Samani).	121
A.2.5. Evapotranspiración diaria.	122
A.2.6. Precipitación en la campaña 2016 – 2017	123
A.2.7. Ejemplo del uso de Amelia II.	124
A.2.8. Precipitación y evapotranspiración en la campaña 2016 – 2017.	124
A.2.9. Precipitación útil en la campaña 2016 – 2017	125
A.3.1. Informes de uso.	128
A.3.2. Inspeccionar por días	129
A.3.3. Consumo en vivo	129

Índice de tablas

2.1. Diferencias entre SiP y SoC	25
3.1. Ejemplo de requisito del sistema	35
3.2. Requisito de sistema – Requisito funcional 01	35
3.3. Requisito de sistema – Requisito funcional 02	36
3.4. Requisito de sistema – Requisito funcional 03	36
3.5. Requisito de sistema – Requisito funcional 04	36
3.6. Requisito de sistema – Requisito funcional 05	36
3.7. Requisito de sistema – Requisito funcional 06	37
3.8. Requisito de sistema – Requisito funcional 07	37
3.9. Requisito de sistema – Requisito funcional 08	37
3.10. Requisito de sistema – Requisito funcional 09	37
3.11. Requisito de sistema – Requisito funcional 10	38
3.12. Requisito de sistema – Requisito funcional 11	38
3.13. Requisito de sistema – Requisito funcional 12	38
3.14. Requisito de sistema – Requisito no funcional 01	39
3.15. Requisito de sistema – Requisito no funcional 02	39
3.16. Requisito de sistema – Requisito no funcional 03	39
3.17. Requisito de sistema – Requisito no funcional 04	39
3.18. Requisito de sistema – Requisito no funcional 05	40
3.19. Requisito de sistema – Requisito no funcional 06	40
3.20. Requisito de sistema – Requisito no funcional 07	40
3.21. Requisito de sistema – Requisito no funcional 08	41
3.22. Requisito de sistema – Requisito no funcional 09	41
3.23. Requisito de sistema – Requisito no funcional 10	41

3.24. Ejemplo de caso de uso	44
3.25. Escenario 1 – Caso de uso 1	44
3.26. Escenario 1 – Caso de uso 2	45
3.27. Escenario 1 – Caso de uso 3	45
3.28. Escenario 1 – Caso de uso 4	46
3.29. Escenario 2 – Caso de uso 1	46
3.30. Escenario 2 – Caso de uso 2	47
3.31. Escenario 2 – Caso de uso 3	47
3.32. Escenario 2 – Caso de uso 4	47
3.33. Escenario 2 – Caso de uso 5	48
3.34. Escenario 2 – Caso de uso 6	48
3.35. Escenario 2 – Caso de uso 7	48
3.36. Escenario 2 – Caso de uso 8	49
3.37. Escenario 3 – Caso de uso 1	49
3.38. Escenario 3 – Caso de uso 2	49
3.39. Escenario 3 – Caso de uso 3	50
5.1. Resumen de las pruebas	80
6.1. Evolución del proyecto	82
6.2. Coste de los materiales	85
6.3. Otros costes	85
6.4. Salarios de los empleados	86
6.5. Coste aplicando el IVA	86
A.2.1. Radiación extraterrestre a 40° de latitud	119
A.2.2. Datos extraídos de AEMET	120
A.2.3. Resultados precipitación, evapotranspiración y lluvia útil	125
A.2.4. Resultados finales en la campaña 2016 – 2017	126

Capítulo 1

Introducción y Objetivos

El primer capítulo del presente documento tiene como objetivo introducir al lector en el entorno de este proyecto. Así pues, la primera sección explica el contexto donde se enmarca el trabajo. Acto seguido, se expondrán los problemas existentes en este ámbito para fijar, en la siguiente sección, los objetivos que se pretenden alcanzar con la realización de este proyecto. Posteriormente, el entorno socioeconómico y finalmente, se presentará una estructura del documento indicando los capítulos que lo formarán junto con una descripción detallada del contenido en cada uno de ellos.

1.1. Contexto

Antes, hoy y mañana, el ser humano ha tenido la necesidad de construir y desarrollar nuevos objetos que faciliten su trabajo y mejoren su calidad de vida [1]. Entre los aspectos más notables en el área de las comunicaciones y la computación, destaca la aparición de los ordenadores personales y el acceso a internet. Estos dos elementos permiten que gran parte de empresas y organizaciones puedan intercambiar una gran cantidad de información en un tiempo muy corto. Además, la aparición de los dispositivos móviles inteligentes supuso un cambio muy drástico en la sociedad actual ya que la comunicación

entre personas se ha facilitado radicalmente pasando de las cartas o las cabinas de teléfono a hacer llamadas desde un terminal de bolsillo. De hecho, otro paso para mejorar la efectividad de los trabajos se consigue gracias a la aparición del Internet de las Cosas, no solo teléfonos, sino miles de millones de dispositivos electrónicos (como televisiones, electrodomésticos, sensores...) están conectados a Internet [2]. La posibilidad de compartir información de manera instantánea en cualquier parte del mundo elimina por completo las barreras que impedían el acceso a la comunicación y la información.

Desde la comunicación y la información, aparecen las Tecnologías de la Información y la Comunicación (o TIC) que son todos aquellos recursos, herramientas y programas que se utilizan para procesar, administrar y compartir la información mediante diversos soportes tecnológicos [3]. Forman parte de la mayoría de los sectores como la educación, robótica, el empleo, la salud o la alimentación [4]. Por ejemplo, en el punto de la alimentación, la agricultura y la ganadería están abiertas a este mundo de nuevas posibilidades ya que es necesario satisfacer la demanda de alimento de la población mundial que está en constante crecimiento. Muchas actividades como la siembra, la gestión del rebaño, la fertilización de las tierras o el mantenimiento de la máquina agrícola se pueden automatizar y aumentar la producción sin tener que malgastar grandes cantidades de recursos que son esenciales diariamente [5].

1.2. Motivación

Para continuar con el mundo de la agricultura y la ganadería, el siguiente proyecto se centrará en un municipio de Toledo, llamado Corral de Almaguer, a 100 kilómetros de Madrid. Su economía se basa principalmente en la agricultura y la ganadería. En cuanto a la agricultura, se fomenta mucho la recolección de las uvas para su procesamiento en vino y de las aceitunas en aceite de oliva.

No obstante, no todo es recoger el fruto de la planta. Se necesita trabajar la tierra y la planta todo el año para que al final de este, se tenga mucho fruto y calidad. Una de las cosas más importantes que necesitan las plantas y las personas es el agua. Es el principio básico para vivir y sin ella, el tiempo de supervivencia es menor. En el caso de las plantas, morirían y su fruto se perdería. Para ello, una solución que se está utilizando hoy en día, es la extracción de aguas subterráneas que se encuentran en acuíferos a diferentes niveles

bajo tierra. Gracias a los motores de gasoil o las placas solares, se obtiene energía suficiente para que las bombas que se encuentran debajo de la superficie extraigan el agua necesaria para uso urbano y/o agrícola.

El agua subterránea de los acuíferos se extrae mediante unas bombas de extracción de diferentes potencias (en función de los metros en los que esté el acuífero) y la energía que se utiliza para hacer funcionar las bombas es o bien solar (gracias a una placa solar fija en el suelo), o bien por motores de gasoil que no están fijos. Alejándose del tema medioambiental, muchos agricultores todavía optan por el motor de gasoil ya que con un motor pueden ir a diferentes pozos y regar todo el tiempo que consideren ya que la placa solar está fija en el suelo y depende de la luz del sol para funcionar.



Figura 1.1: Ejemplo de motor de gasoil – DEUTZ 33 kVA [6]



Figura 1.2: Ejemplo de panel solar destinado al riego [7]

Así pues, un aspecto que preocupa a los agricultores es la seguridad de su maquinaria agrícola, en especial de los motores de gasoil, debido a que en los últimos años se han

producido robos totales de estas mismas [8]. Muchas de las veces las máquinas son irrecuperables ya que se han vuelto a vender de manera ilegal o se han desguazado para conseguir los metales de estas mismas [8]. Este es un problema grave ya que estos motores tienen un valor sustancial [8] y requiere de una inversión inicial por parte del agricultor para poder utilizarlos.

Para resolver estos problemas, que son comunes en la actualidad, es vital el uso de las Tecnologías de la Información y la Comunicación. El empleo del Internet de las Cosas y las redes de sensores inalámbricas ayudaría a automatizar estas tareas y a impulsar el desarrollo de las zonas rurales conectándolas con el futuro. Un futuro que se llama sociedad digital.

1.3. Objetivos

El objetivo general de este trabajo fin de grado consiste en diseñar e implementar un sistema de monitorización del riego de cultivos (especializado en la vid) y equipos agrícolas (motores de gasoil), utilizando sensores inalámbricos capaces de enviar información para que el sistema sea capaz de procesar y tomar decisiones a partir de la misma.

La monitorización y la toma de decisiones van orientadas a cumplir con dos metas fundamentales:

- ❖ Evitar gastar más agua de la que se necesita a partir del control del agua para un riego óptimo.
- ❖ Evitar el robo del motor de gasoil con un control de seguridad.

1.3.1. Objetivos específicos

Para conseguir el objetivo general de este trabajo, se deben cumplir con los siguientes objetivos específicos:

- Estudiar el uso y despliegue de redes de sensores en el sector agrícola.
- Explorar los sensores disponibles y aprender cómo funciona.

- Desarrollar un nodo, utilizando Raspberry Pi, que incorpore los sensores necesarios, para cada situación particular, tanto de control de riego como de la seguridad.
- Incorporar, a dichos nodos, capacidad de comunicación tanto local como remota.
- Cifrar la información transmitida entre los sensores y la estación central usando un estándar que proporcione un nivel de seguridad alto.

1.4. Medios utilizados

1.4.1. Software

A continuación, se lista el software que fue útil para el desarrollo del proyecto:

- ❖ **Python:** Es el lenguaje principal de programación utilizado en el proyecto gracias a su facilidad para conseguir información de los sensores y tratarla.
- ❖ **Raspbian:** Sistema operativo de la Raspberry Pi.
- ❖ **MobaXterm:** Es la terminal utilizada con la que se pueden realizar conexiones *SSH* del ordenador a la Raspberry Pi.
- ❖ **RStudio:** El entorno de desarrollo en R que sirvió para el procesamiento de datos en el anexo 2.
- ❖ **R:** Lenguaje de programación útil para la manipulación de datos. Se utilizó en el anexo 2.
- ❖ **Hologram Cloud:** Lugar donde se gestionan los datos que se han utilizado en el módem conectado a la Raspberry Pi. Además, desde ahí se envían *SMS* y correos electrónicos de alerta al usuario.
- ❖ **Thingspeak:** Un sitio web donde se enviarán los datos de la humedad en tiempo real. Se podrá ver esta información gráficamente.
- ❖ **Microsoft Office 365 – Microsoft Word:** Utilizado para la documentación del proyecto.
- ❖ **Microsoft Office 365 – Microsoft Excel:** Utilizado para hacer el diagrama de Gantt.
- ❖ **Draw.io:** Página web útil para realizar los diagramas de flujo y otros esquemas del documento.

1.4.2. Hardware

En cuanto a *Hardware*, a continuación, se mostrarán aquellos dispositivos que han hecho posible el proyecto:

- ❖ **Raspberry Pi 3 Model B:** Como cualquier otro ordenador, la Raspberry Pi dispone de una serie de puertos: USB, HDMI, Ethernet, Audio Jack, tarjeta de memoria microSD, módulo para conectar una pantalla táctil, una cámara, Bluetooth y WiFi en su última versión. La placa monta un procesador ARM 64-bits de cuatro núcleos a 1.2 GHz y con un 1 GB de memoria RAM. Además, cuenta con 40 pines de GPIO destinados al uso de los sensores.



Figura 1.3: Raspberry Pi 3 Model B

- ❖ **Higrómetro FC-28:** Es un sensor que mide la humedad del suelo por la variación de su conductividad. Son ampliamente empleados en sistemas automáticos de riego para detectar cuando es necesario activar el sistema de bombeo.

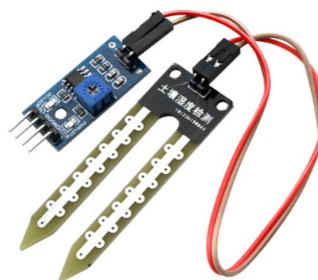


Figura 1.4: Higrómetro FC-28 [9]

El FC-28 viene acompañado con una placa de medición estándar que permite obtener la medición como valor analógico o como una salida digital, activada cuando la humedad supera un umbral. En el caso analógico, los valores obtenidos van desde 0 (sumergido en agua), a 1023 (en el aire o suelo muy seco). Sin embargo, en la salida digital, cuando la humedad supera un umbral que se ajusta con el potenciómetro de la Figura 1.5, la salida se dispara. Se obtendrá una señal LOW si el suelo no está húmedo, y HIGH si el FC-28 supera el umbral [10].

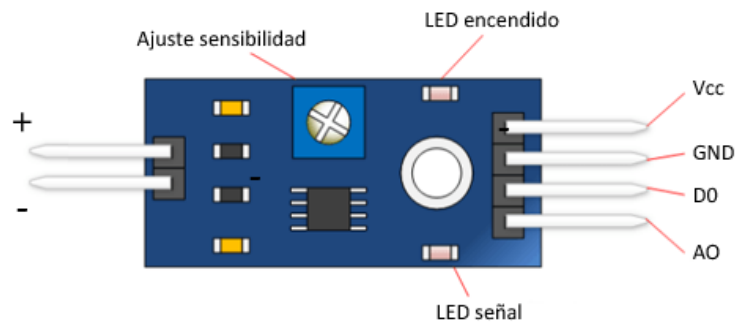


Figura 1.5: Vista con detalle de la placa de medición estándar [10]

- ❖ **MCP3008:** Es un convertidor analógico a digital de 10 bits diseñado por la empresa Microchip. Es muy útil ya que la Raspberry Pi no tiene pines de entrada analógicos. El ADC se colocaría entre la Raspberry PI y el FC-28 y permitiría que la Raspberry Pi pudiese leer los valores exactos del sensor de humedad FC-28.

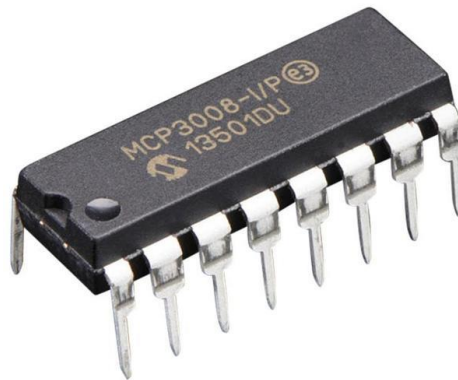


Figura 1.6: Ejemplo del ADC MCP3008 [11]

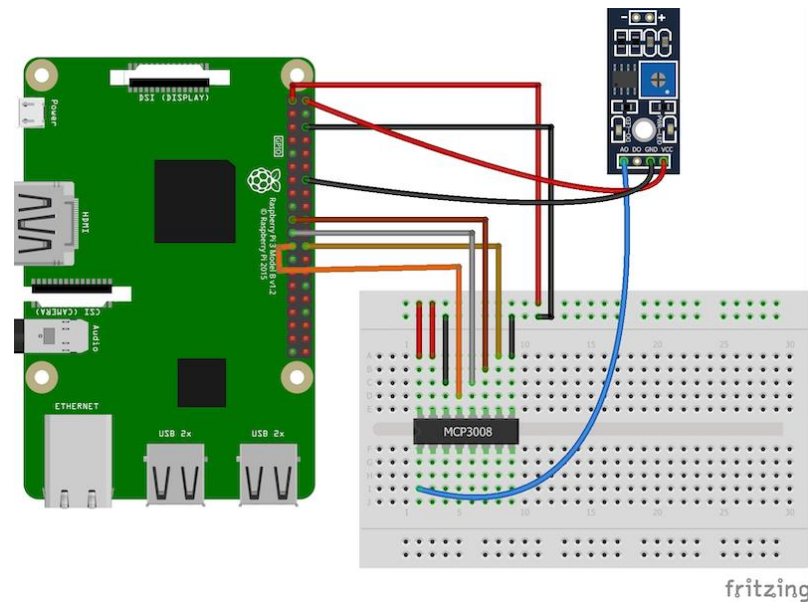


Figura 1.7: Esquema de Raspberry Pi, el MCP3008 y el FC-28 [12]

- ❖ **HC-SR501:** Es un sensor infrarrojo pasivo o *PIR* que detecta variaciones de las radiaciones infrarrojas del medio ambiente que cubre como el calor del cuerpo humano o animales. Su sensor tiene una apertura de 110° y con un potenciómetro, se puede calibrar una distancia de detección de 3 a 7 metros.



Figura 1.8: Sensor *PIR* HC-SR501 [13]

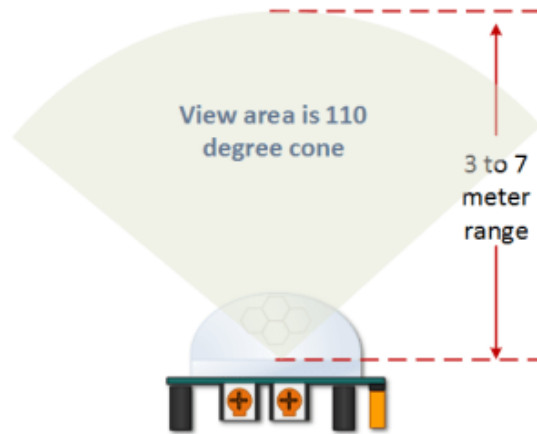


Figura 1.9: Esquema de detección del sensor HC-SR501 [14]

Como se puede ver en la Figura 1.9, el *PIR* tiene un segundo potenciómetro que sirve para calibrar cada cuándo se recogen datos del sensor.

Además, el sensor del *PIR* tiene dos ranuras con un material sensible a las radiaciones infrarrojas. Cuando ambas ranuras detectan la misma radiación, el sensor está en reposo, mientras que cuando las ranuras detectan diferente radiación, se detecta una variación de movimiento térmico.

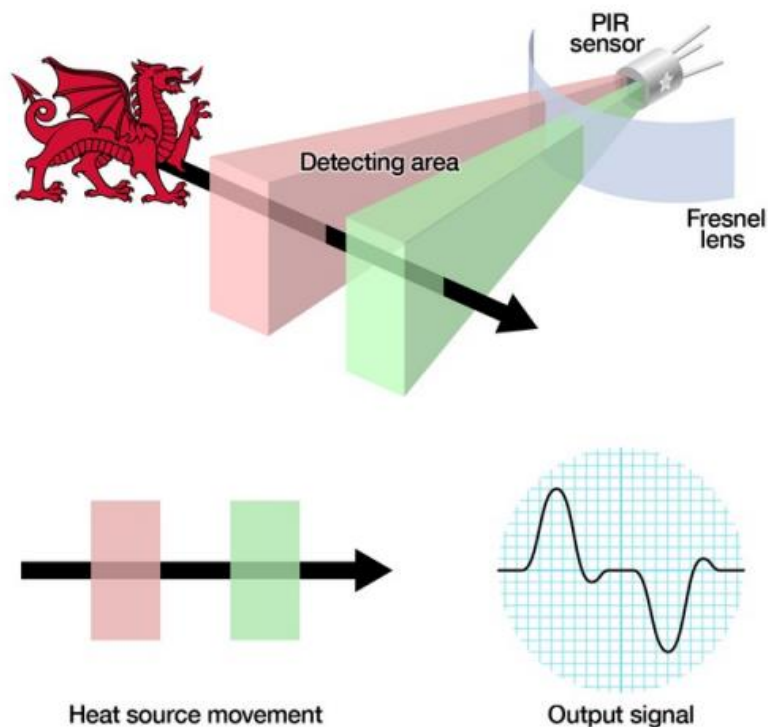


Figura 1.10: Esquema de detección de movimiento [14]

- ❖ **MPU6050:** Es un dispositivo integrado de seguimiento de movimiento de 6 ejes que combina un giroscopio de 3 ejes y un acelerómetro de 3 ejes. Utiliza comunicación I2C, es decir, comunicación síncrona con cable para el reloj (SCL) y otro para los datos (SDA). El maestro sería la Raspberry Pi el cual crea la señal de reloj. Los ejes son X, Y se pueden ver en la Figura 1.11 mientras que el eje Z es el ortogonal a los dos ejes anteriores.

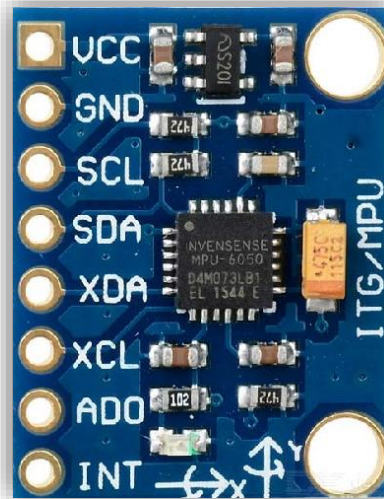


Figura 1.11: Vista general del giroscopio y acelerómetro MPU6050 [15]

- ❖ **Hologram SIM:** Es una tarjeta SIM *IoT* segura para una red celular global. Tiene fácil activación ya que no es necesario pedir permisos a las compañías de telefonía, sino que todo el proceso ya está gestionado por Hologram. De hecho, la tarjeta utiliza la cobertura del operador que más potencia suministre en el área donde se encuentre.

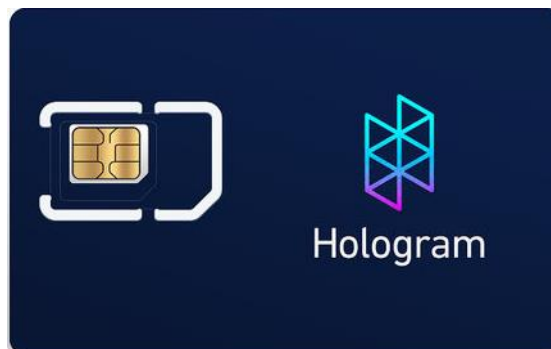


Figura 1.12: Ejemplo tarjeta SIM Hologram [16]

- ❖ **Hologram Nova:** Es el módem recomendado por Hologram para poder tener conexión a la red en cualquier dispositivo. Ya que Raspberry Pi tiene varios USB, Hologram Nova es una opción sencilla de usar.



Figura 1.13: Módem - Hologram Nova [16]

Para su funcionamiento, es necesario insertar la tarjeta SIM (en el formato nano SIM) en el módem y conectarlo directamente en la Raspberry Pi. Después, es necesario instalar el paquete Hologram en la Raspberry Pi con el siguiente comando:

```
curl -L hologram.io/python-install | bash
```

Figura 1.14: Comando de instalación [16]

Una vez hecho, Hologram Nova ya estará disponible para cualquier actividad en la Raspberry Pi.

- ❖ **Batería portátil:** Para poder mantener la Raspberry Pi encendida a larga distancia es necesario tener una batería portátil. En este proyecto, se ha utilizado una batería de 10000 mAh que da lugar a 12 horas de funcionamiento aproximadamente.

1.5. Entorno socioeconómico

El objetivo principal del Internet de las Cosas es hacer del mundo un lugar más eficiente y, por consiguiente, conseguir un ahorro energético y económico. Potenciando esta tecnología, cada vez se incidirá más en su uso en el ámbito empresarial y en la modernización de otros sectores como el agrario ya que mediante el uso de sensores, se optimizan muchas tareas como el regado y cuidado de las plantas.

Además, es necesario tener un control entre los recursos del planeta junto con las condiciones materiales de vida de las personas. Por ejemplo, en la actualidad, el agua y su mantenimiento están siendo cada vez más los objetos principales de los debates en torno al desarrollo sostenible.

Sin embargo, muchos trabajadores cuestionan que esta evolución a la sensorización y la automatización suponga un problema en sus propios trabajos. Todo cambio tecnológico parece difícil de asumir, pero la población de hoy en día ha demostrado que es muy fácil adaptarse a las nuevas tecnologías. Algunos ejemplos se pueden encontrar en los *Smartphones* o en el sector automovilístico en los que se encuentran cada vez más funciones que facilitan una conducción más segura y cómoda.

1.6. Estructura de la memoria

Sin tener en cuenta el presente capítulo, a continuación, se explicará cuáles son los siguientes capítulos del documento:

- ❖ **Capítulo 2. Estado del arte:** en este apartado se explicarán los conceptos actuales en los que se basará el proyecto, como el de las redes de sensores (*Wireless Sensor Networks*) y el Internet de las Cosas (*Internet of Things*), la tecnología relacionada con la agricultura y finalmente, se presentarán con detalle los dispositivos y sensores que van a ser útiles en este proyecto.
- ❖ **Capítulo 3. Análisis y diseño:** Se explicará cuál deberá ser el diseño del proyecto para estableciendo todos los requisitos del sistema que se proponen. Será un modelo lógico para entender cómo debería funcionar el sistema completo.
- ❖ **Capítulo 4. Implementación:** En el aspecto práctico, se explicará cómo se realizará un prototipo del proyecto y qué adaptaciones se harán para que siga funcionando de acuerdo con las especificaciones mínimas requeridas.
- ❖ **Capítulo 5. Evaluación:** Se evaluarán los requisitos del sistema y se valorará si se han cumplido correctamente. En el caso de que algo no se haya cumplido, se expondrá la justificación.
- ❖ **Capítulo 6. Planificación y presupuesto:** En esta sección, se verá cómo ha sido la planificación del proyecto y cuál va a ser el coste de cada una de las partes del prototipo, distribuyendo los gastos a lo largo del desarrollo del proyecto.

- ❖ **Capítulo 7. Marco Regulator:** De mostrarán aquellas normas o leyes que tengan relación con este proyecto y se explicará por qué se cumplen estas mismas.
- ❖ **Capítulo 8. Conclusiones y líneas futuras:** En este apartado, se mostrarán los resultados obtenidos de la simulación del prototipo, así como propuestas de mejora para un mejor funcionamiento en el futuro.
- ❖ **Anexos:** serán importantes para comprender completamente la situación del proyecto (anexos 2 al 4). Además, en el anexo 1, se incluye un resumen de las diferentes partes del trabajo en inglés. Por otro lado, existe un glosario de acrónimos y definiciones en el anexo 5.

Capítulo 2

Estado del arte

Antes de la presentación del diseño y la implementación del prototipo, se van a explicar una serie de conceptos que son esenciales para el trabajo. En primer lugar, se abordará el concepto de Internet de las Cosas para explicar a continuación en qué consiste la Red de sensores inalámbricas. Finalmente, se presentarán los dispositivos que fueron utilizados para cumplir con las expectativas del proyecto.

2.1. Internet de las Cosas

En el momento en el que apareció Internet, éste empezó a evolucionar de una manera muy veloz, desde los antiguos módems hasta las eficientes líneas de fibra óptica actuales. Hoy en día, múltiples dispositivos electrónicos como los móviles, los Smart TV, los Smartwatch... hasta los frigoríficos están conectados a Internet [17]. ¿A qué se debe esto? El tema que engloba esta idea se llama Internet de las Cosas.

El Internet de las Cosas (del inglés, *Internet of Things* o *IoT*) se refiere a un sistema de dispositivos que están interconectados de manera digital con Internet. Son objetos que tienen identificadores únicos y son capaces de transmitir datos a través de una red [18], es decir, en el IoT todas las comunicaciones son Máquina a Máquina (M2M).



Figura 2.1: Internet de las Cosas [19]

El objetivo del *IoT* es facilitar la calidad de vida y a su vez, el modo de vivir de la sociedad. Es clave para administrar eficientemente los recursos que se tienen actualmente. Actualmente, se están realizando proyectos de *IoT* en los que se comprueba la mejora de todo tipo de análisis, recopilación y distribución de datos debido a la enorme capacidad para ello [20, 21].

El concepto de internet de las cosas fue propuesto en 1999 por Kevin Ashton quien usó por primera vez esta expresión en el grupo de investigación Auto-ID Labs. Apareció gracias al estudio de los sistemas por radiofrecuencia, también conocidos como *RFID*, y en las nuevas tecnologías de detección de sensores [22, 23]. Desde entonces, se utiliza de forma normal para referirse a este sistema de conexión de dispositivos a Internet [24].

Con el crecimiento del *IoT*, los objetos o *cosas* inteligentes ofrecen millones de datos provenientes del aumento del número de dispositivos conectados entre sí, ya sea a través de Internet o microchips. Es aquí donde aparece el concepto de *Big Data* y todos esos datos son gestionados por las empresas a través de una Analítica de Datos Masivos [25].

De hecho, en 2018 se llegó a alcanzar alrededor de 17.000 millones de dispositivos conectados a Internet y se prevé que, en 2020, el número se incremente a 20.000 millones de “cosas” conectadas [26]. Además, se calcula que todo ser humano está rodeado, al menos, por un total de aproximadamente 1000 a 5000 objetos (conectados o no a Internet)

[27]. En España, como se puede ver en la Figura 2.3, se puede observar que 1 de cada 4 españoles tienen 5 o más dispositivos conectados a Internet.

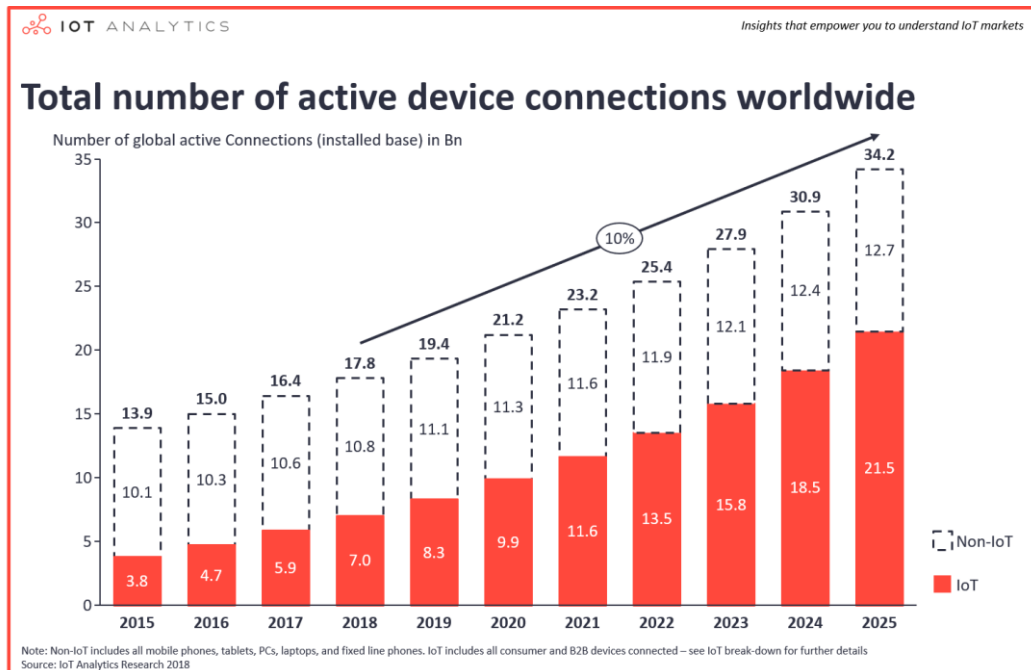


Figura 2.2: Número de dispositivos conectados a Internet en el mundo [28]

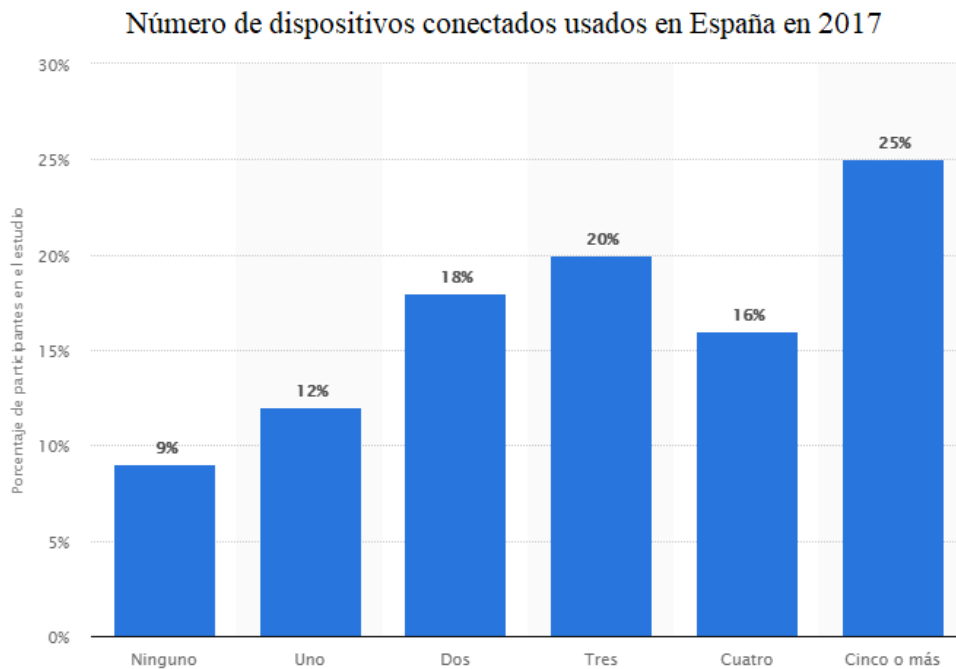


Figura 2.3: Número de dispositivos conectados usados por persona en España [29]

2.1.1. Evolución de las tecnologías con el Internet de las Cosas

Las tecnologías relacionadas con *IoT* se han incrementado en los últimos años. Gracias a la constante innovación, no se puede tener una visión exacta de cómo evolucionará en el tiempo. Sin embargo, hay muchas tendencias tecnológicas que ayudan a ver esa evolución:

- ❖ **Mejora de la Inteligencia Artificial:** Cuando el *IoT* esté funcionando al máximo rendimiento, la Inteligencia Artificial debe estar preparada para recibir una gran cantidad de datos para analizar. Sobre todo, debe mejorar en el Aprendizaje Automático o *Machine Learning* porque para analizar este volumen de datos, es necesario que se pueda entrenar y aprender por sí sola [30].
- ❖ **Mejora de las plataformas móviles:** Volviendo al concepto de *IoT* y aplicando esta idea, las plataformas móviles se convertirán en un importante sistema de gestión para los dispositivos *IoT* [30].
- ❖ **Impacto en las relaciones con los clientes:** La experiencia del cliente se verá influenciada positivamente por la mayor personalización de los productos.
- ❖ **Crecimiento del pequeño negocio en los próximos años:** De acuerdo con Forrester Research [31], el *IoT* se convertirá en una herramienta fundamental para permitir la conexión entre los negocios pequeños y los clientes.
- ❖ **Mayor fragmentación:** A medida que vaya creciendo el *IoT*, también se volverá más fragmentado. Es necesario, que haya una mejora e investigación constante a pesar de que varias compañías lideren el impulso de los estándares y certificaciones de dispositivos *IoT*.
- ❖ **Seguridad:** Al igual que en el punto anterior, a medida que el *IoT* crezca, la cantidad de datos y dispositivos será mayor y esto indicará un mayor número de brechas susceptibles de ataques. Cuanto más complejo sea la red, más protocolos de seguridad se deben aplicar [32].
- ❖ **Mayor inversión y gasto:** La Corporación de Datos Internacional (*IDC*) predice que, en 2023, el gasto mundial en *IoT* alcanzará 1,1 billones de dólares [33]. Esto subraya la importancia de mejorar la experiencia del cliente en casi todas las industrias.

2.1.2. Aplicaciones del Internet de las Cosas

Como se ha introducido anteriormente, el *IoT* está cambiando la relación de las personas con el mundo físico. A continuación, se verán las aplicaciones más significativas en las que el *IoT* tiene un rol importante:

- ❖ **Agricultura y ganadería inteligente:** El *IoT* tiene capacidad para transformar el mundo. Como se ve en los demás puntos, el *IoT* afecta a industrias más eficientes, la automoción, las *Smart Cities*... Sin embargo, la aplicación del *IoT* en la agricultura podría tener el mayor impacto. Es la aplicación fundamental para las ciudades más rurales donde se encuentra principalmente el sector primario.

De acuerdo con Beecham Research y la Organización de las Naciones Unidas para la Alimentación y la Agricultura, para mantener el ritmo con el crecimiento de la población mundial se deberá incrementar la producción agrícola en un 70% para 2050 [34] ya que la población para entonces será de 9600 millones de personas.

La agricultura inteligente basada en tecnologías *IoT* permitirá a los agricultores mejorar la productividad sin desperdiciar los recursos, desde la cantidad de fertilizante utilizado hasta el número de viajes realizados por vehículos agrícolas [35]. En cuanto a sus aplicaciones usando *IoT*, se puede destacar la automatización del riego, la erosión del suelo, la evaluación de salud de los cultivos y la fumigación automática de cultivos. En los dos últimos, se aprecia la existencia de drones agrícolas que realizan estas dos tareas de árbol en árbol [36].



Figura 2.4: Aplicaciones del *IoT* en la agricultura



Figura 2.5: Uso del dron para monitorización de una viña en Argentina [37]

- ❖ **Smart Cities:** En el año 2018, el 23% de todos los proyectos en el mundo del *IoT* estaban dedicados a la mejora de las ciudades inteligentes [38]. Ya que el número de personas en el mundo está creciendo continuamente, se necesita que se mejore la infraestructura en todas las ciudades como en la distribución del agua, el transporte automático, en los sistemas de gestión de la energía, la vigilancia inteligente, la seguridad urbana, la monitorización del medio ambiente o el control del tráfico. Este último punto es muy importante para muchas ciudades ya que hay muchos proyectos destinados al sistema del aparcamiento, control de tráfico, aplicaciones de bicicletas o patines eléctricos o la mejora del transporte público. Todos estos proyectos ayudarían a reducir un problema muy presente en las ciudades, la contaminación del aire [39].
- ❖ **Domótica o Smart Home:** Automatizar las tareas cotidianas aporta un plus de confort que proporciona más tiempo libre para dedicarlo a otras cosas. Uno de los beneficios más señalados es el aumento de seguridad tras la instalación de alarmas y cámaras en el interior y exterior de la vivienda. Otra de las razones importantes, es el ahorro energético que se obtiene cuando se acondiciona las distintas estancias de la casa. Además de repercutir en la economía del usuario, también lo hace en el medio ambiente ya que muchas marcas de electrodomésticos dedican un gran presupuesto en la investigación para crear dispositivos más eficientes y respetuosos con el entorno. Además, todo pasaría a controlarse con el teléfono inteligente ya que actualmente, se debe activar cada electrodoméstico de manera presencial [40]. Gracias al *IoT*, se pueden ver las alertas recibidas por cada electrodoméstico y actuar en función de la situación.



Figura 2.6: Mejora de la domótica con el *IoT* [41]

- ❖ **Conducción Autónoma:** La tecnología digital automotriz se ha centrado en optimizar las funciones internas de los vehículos. Sin embargo, ahora, la atención está creciendo hacia la mejora de la experiencia del automóvil. El objetivo fundamental es la mejora de la percepción del medio ambiente con los sensores *IoT* para permitir una conducción más segura [42].

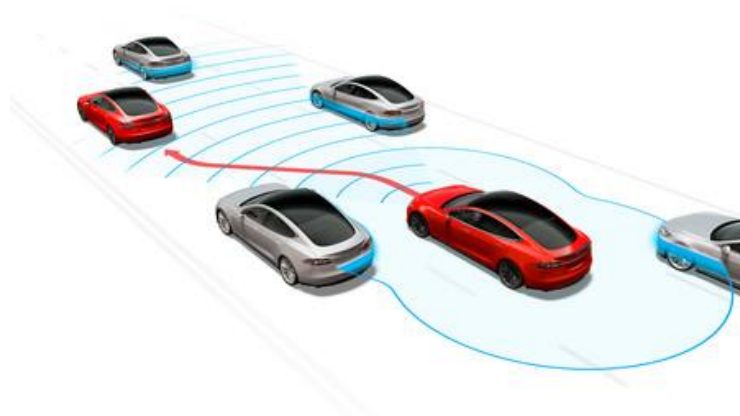


Figura 2.7: Ejemplo del uso de sensores para conducción autónoma [43]

- ❖ **Sector Sanitario:** También llamado el Internet de las Cosas Médicas (*Internet of Medical Things, IoMT*). Son dispositivos que se usan en las áreas de la atención médica y la salud [44]. Los dispositivos médicos capturan los datos y de forma inalámbrica los envían a otros dispositivos o a la nube. Estos datos son analizados y desencadenan la acción pautada en los protocolos. Finalmente, los médicos utilizan la información para la toma de decisiones. Sus beneficios son la disminución de los costos en asistencia en salud, reduce los errores de diagnóstico

al recibir más datos, la experiencia del paciente mejora considerablemente y ayuda a los pacientes a adherirse a los medicamentos [45].



Figura 2.8: Internet de las Cosas Médicas [44]

❖ **Internet Industrial:** Al igual que en el sector sanitario, existe el Internet Industrial de las Cosas (*IIoT*). Son los dispositivos que están destinados al uso de la Industria. Los beneficios el *IIoT* son las mejoras en la gestión de instalaciones ya que los dispositivos pueden usarse para anticipar problemas originados por humedad o vibraciones, la logística porque la industria está conectada en tiempo real con el cliente y proveedores, y la medición inteligente porque los dispositivos pueden controlar el consumo de recursos y así reducir el gasto de manera significativa [46].



Figura 2.9: Internet Industrial de las Cosas [47]

2.2. Redes de sensores inalámbricas

Las redes de sensores inalámbricas (del inglés, WSN, *Wireless Sensor Networks*) están formadas por nodos sensores o motas, que son dispositivos autónomos distribuidos a lo largo de un área de interés y cuyo objetivo es monitorizar parámetros físicos o ambientales tales como temperatura, sonido, vibraciones, presión, movimiento o agentes contaminantes [48]. Estos dispositivos trabajan de manera colaborativa para recoger los datos y enviarlos a una pasarela o *Gateway* para que este los transmita a la red exterior (sea un servidor o la nube).

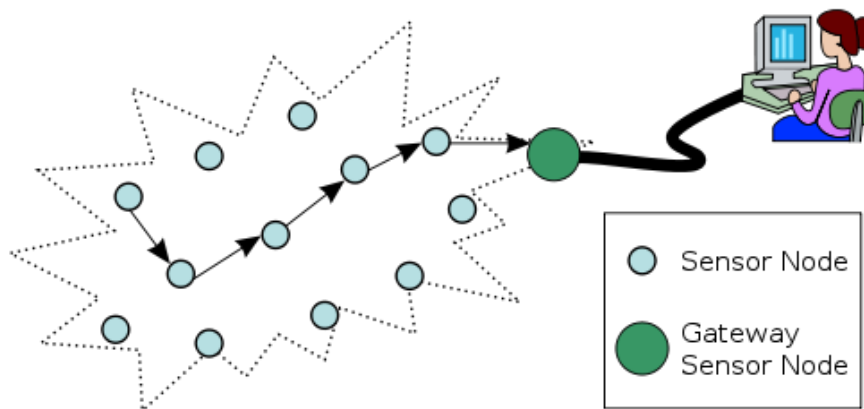


Figura 2.10. Ejemplo de red de sensores inalámbrica [48]

2.2.1. Usabilidad de las Redes de Sensores Inalámbricas

A continuación, se presentarán las características más importantes de las redes de sensores que se han habilitado como una herramienta potencial en el dominio agrícola.

- i. **Capacidad de toma de decisiones inteligente:** Esta característica mejora la eficiencia de la energía de toda la red de sensores y, por lo tanto, el tiempo de usabilidad de la red aumenta. De hecho, muchos nodos o motas colaboran entre ellos y colectivamente toman una decisión definitiva [49].
- ii. **Configuración de topología dinámica:** Para maximizar la vida útil de la red, la topología de la red se configura de tal manera que el número de nodos activos sea mínimo [50]. Los nodos sensores se mantendrán en el “modo de suspensión” para conservar la energía de la batería.

- iii. **Tolerancia al fallo:** Una de las dificultades más comunes en la implementación de las redes de sensores es que los nodos sean propensos a fallar [51]. Esto implica que la red se particiona y el rendimiento se ve afectado. Para ello, los nodos se organizan reconfigurando la topología dinámica.
- iv. **Conocimiento del contexto:** En base a la información que reciben los sensores, estos adquieren el conocimiento suficiente sobre el ambiente que les rodean. Las soluciones que los sensores toman posteriormente dependen del contexto [52].
- v. **Escalabilidad:** Los protocolos WSN están diseñados para implementarse en cualquier red, independientemente de su tamaño y número de nodos. Esta característica amplía el potencial del WSN para numerosas aplicaciones [53].
- vi. **Heterogeneidad de los nodos:** Las redes de sensores son heterogéneas cuando tienen diferentes tipos de sensores en términos de energía, ancho de banda y memoria [54].
- vii. **Tolerancia contra los fallos de comunicación en condiciones ambientales adversas:** Debido a la amplia gama de aplicaciones en ambientes agrícolas abiertos, las redes de sensores sufren los efectos de las condiciones ambientales adversas. Para ello, existe una pila de protocolos de WSN que incluye técnicas para resistir este tipo de fallos debido a los efectos medioambientales [55].
- viii. **Modo de funcionamiento autónomo:** Una característica importante de las WSN es su modo de funcionamiento autónomo [56]
- ix. **Seguridad de la información:** Las redes de sensores llevan información sin formato sobre los parámetros del campo. Para garantizar la seguridad de la información detectada y restringir a los usuarios no autenticados, las WSN proporcionan mecanismos de control de acceso y detección de anomalías [57].

2.2.2. Arquitectura de la red de sensores en la agricultura

Dependiendo del tipo de aplicación que se necesita, se debe buscar una arquitectura u otra. Las arquitecturas de red se clasifican en varias categorías como el movimiento de los nodos sensores, en sus tipos de sensores en términos de heterogeneidad, o en la jerarquía [58]. En el primer punto, el movimiento de los nodos sensores, se pueden identificar las diferentes arquitecturas:

- ❖ **Arquitectura estacionaria:** Los nodos sensores están desplegados en una posición fija en el período en el que dure la aplicación. Por ejemplo, es ideal para sistemas de monitorización de riego o control del uso de fertilizantes.
- ❖ **Arquitectura móvil:** Los nodos sensores cambian su posición dependiendo del tiempo. Por ejemplo, una aplicación basada en esta arquitectura sería el control de tractores autónomos.
- ❖ **Arquitectura híbrida:** Sería la combinación entre nodos estacionarios y móviles. Por ejemplo, sería aplicable en sensores del campo, equipos agrícolas móviles, usuarios de teléfono móvil y transporte de ganado.

En cuanto a los tipos de sensores, se pueden encontrar las siguientes categorías:

- ❖ **Arquitectura homogénea:** Se compone de dispositivos equipados con sensores de potencial similar. Este tipo de marco se usa normalmente en aplicaciones basadas en implementaciones no planificadas. Sin embargo, este tipo de arquitectura carece de variedad en términos de hardware de comunicación. En consecuencia, los esquemas y protocolos de comunicación están diseñados teniendo en cuenta esta limitación. Un ejemplo de aplicación es la recolección de datos sobre la cantidad de nutrientes en el suelo.
- ❖ **Arquitectura heterogénea:** En este tipo de arquitectura, existen varios tipos de nodos sensores y dispositivos. Estos dispositivos varían en términos de potencia de cálculo, memoria, capacidad de detección y unidades de transceptor. Por ejemplo, en cualquier aplicación de gestión de riego, los nodos de sensores en campo comunican su información detectada a la pasarela, que de nuevo transfiere la información a un usuario remoto (comunicación por radio frecuencia o GSM).

Finalmente, en términos de jerarquía, las arquitecturas se clasificarían en las siguientes categorías:

- ❖ **Arquitectura de un solo nivel:** es común en aplicaciones de pequeña escala. En este tipo de arquitectura, los dispositivos en el campo y los nodos sensores comunican directamente sus datos a la pasarela, que está colocada cerca del área de aplicación.
- ❖ **Arquitectura multinivel:** Los nodos de sensores en el campo permanecen en el nivel inferior de la jerarquía y forman los grupos básicos o clústeres. Entonces, los siguientes niveles de jerarquía incluyen varios clústeres para llegar a los nodos

de la pasarela. Normalmente, las arquitecturas multinivel consisten en nodos heterogéneos.

2.2.3. Arquitectura de los nodos sensores

En este apartado, en la arquitectura de las motas o los nodos sensores se pueden distinguir dos tipos:

- ❖ **Sistema en paquete** (*System-in-Package, SiP*): Se define como la combinación de dispositivos múltiples que incluyen elementos pasivos (como resistencias y condensadores), montados juntos y manteniendo la disposición de conectar componentes externos posteriormente. Los factores para la selección de los componentes de una mota de este tipo son el procesador, el transmisor-receptor, la memoria, la potencia y el coste.
- ❖ **Sistema en chip** (*System on Chip, SoC*): Sigue una reducción de diseño más específica para cada aplicación. Así pues, minimiza los requisitos de energía y coste del diseño. Proporciona una integración de múltiples núcleos de procesadores programables, unidades de memoria, unidades de entrada/salida y bloques personalizados.

En la tabla siguiente, se pueden observar las diferencias entre ambos sistemas:

Factor	SiP	SoC
Procesador	Pocos y heterogéneos	Múltiple y heterogéneo
Consumo	Alto	Bajo
Coste	Alto	Bajo
Tamaño	Grande	Pequeño
Memoria	En un módulo separado	Integrado

Tabla 2.1: Diferencias entre *SiP* y *SoC* [Elaboración propia, 58]

2.3. Tecnologías y estándares usados en la agricultura

En esta sección, se explicará con detalle las diferentes tecnologías de comunicación inalámbrica y los estándares presentes en las aplicaciones en la agricultura.

2.3.1. Zigbee

Esta tecnología define la red y los protocolos de la capa de aplicación basados en la norma IEEE 802.15.4 y las definiciones de la capa MAC necesarias para diseñar una red inalámbrica de área personal (*Wireless Personal Area Network, WPAN*) utilizando dispositivos habilitados para radio de baja potencia. Admite la comunicación a corta distancia (10-20 metros) a través de redes multinivel y de malla. Su velocidad de datos es relativamente baja entre 20-40 kbps en la banda de frecuencia ISM (Bandas de radio industriales, científicas y médicas [59]) de 868-915 MHz y 250 kbps a 2.4 GHz [60]. No requiere de un *hardware* de alta especificación ya que se anuncian dispositivos con hasta 128 kB de almacenamiento [61]. Es eficiente en energía, coste bajo y confiable por ello es muy utilizada en la agricultura. Se definen tres tipos distintos de dispositivo Zigbee [62]:

- ❖ **Coordinador Zigbee:** Uno por red ya que se encarga de controlar esta y los caminos que deben seguir los dispositivos para conectarse entre ellos.
- ❖ **Router Zigbee:** Interconecta dispositivos separados en la red.
- ❖ **Dispositivo final:** Tiene la funcionalidad necesaria de comunicarse su nodo padre (sea coordinador o router) pero no puede transmitir su información a otros dispositivos. De esta forma, el nodo suele estar en suspensión la mayor parte del tiempo, conservando la energía de las baterías.



Figura 2.11: Ejemplo de módulo de red Zigbee [63]

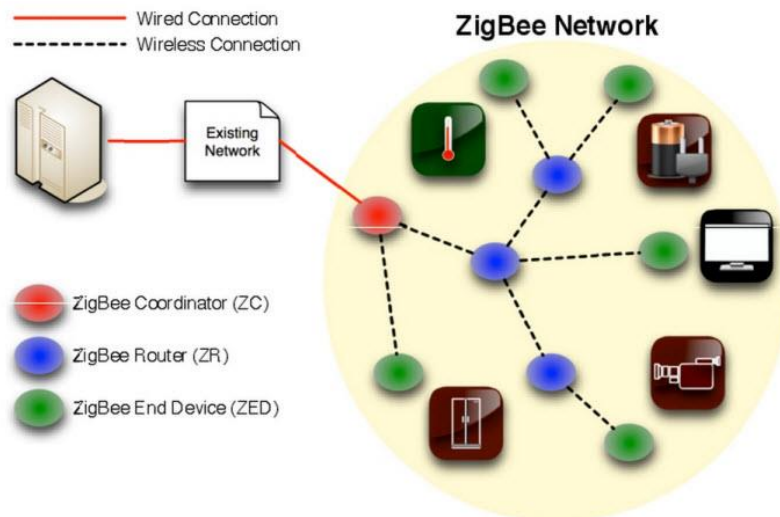


Figura 2.12: Red de Sensores Zigbee [64]

2.3.2. WiFi

WiFi es un estándar de red de área local inalámbrica (WLAN) para el intercambio de información o la conexión a Internet de forma inalámbrica basada en la familia de estándares IEEE 802.11. WiFi ofrece un rango de comunicación adecuado del orden de 20 m (interior) a 100 m (exterior) con una velocidad de transmisión de datos del orden de 2–54 Mbps a una frecuencia de 2,4 GHz de la banda ISM [65, 66].

2.3.3. Redes móviles y su evolución

La red GSM (Sistema global de comunicaciones móviles o *Global System for Mobile Communications*) es un estándar de segunda generación (2G) ya que las comunicaciones empiezan a producirse de un modo completamente digital en comparación con la primera generación de teléfonos. [67, 68]. En comparación con las otras tecnologías, su rango de transmisión de datos se extiende por todo el mundo. Su banda de frecuencia depende del territorio, aunque en Europa se utilizan las bandas de 900 y 1800 MHz y se logra una velocidad de datos de 50-100 kbps [69]. Los teléfonos o estaciones móviles necesitan de una tarjeta SIM para poder utilizar la tecnología GSM, que contiene información sobre el terminal y su usuario [70].

Más adelante, nuevas generaciones aparecieron para mejorar la segunda generación como GPRS, EDGE, UMTS (3G), 4G y 5G. Actualmente, es fácil conectarse a la red vía 3G o

4G ya que el 5G todavía no se ha implementado en todos los lugares. En España, las velocidades de descarga rondan los 6.5 Mbps en 3G, mientras que en 4G rondan los 25 Mbps [71].

Por otro lado, a parte de su uso en comunicación, GSM también puede utilizarse para localizar un dispositivo sin tener que recurrir al uso del GPS. Esto es muy útil para aquellos dispositivos que no tengan GPS. Por lo general, un móvil o dispositivo con GSM está conectado a una o varias torres de telefonía. Mediante un cálculo de triangulación, es posible saber la posición del objeto con un pequeño margen de error que dependerá del número de torres en la zona. El dispositivo localizará las torres más cercanas y calculará el tiempo que tarda la señal de torre a torre y la fuerza de esta [72, 73, 74, 75].

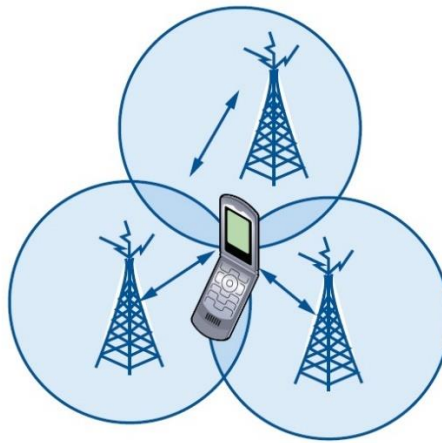


Figura 2.13: Ejemplo de triangulación por GSM para localizar un dispositivo [76]

2.4. Tecnologías utilizadas

En este apartado se explicará qué recursos se han utilizado para hacer que el proyecto funcione. Más adelante, en los capítulos del diseño y la implementación se explicará cual será el papel fundamental de cada uno.

2.4.1. Raspberry Pi

Raspberry Pi es un ordenador de bajo coste desarrollado por la Fundación Raspberry Pi en Reino Unido, del tamaño de una tarjeta de crédito, que se conecta a un monitor, y usa un teclado y ratón estándar. Es capaz de hacer todo lo que puede hacer un ordenador

normal, desde navegar por Internet y reproducir videos hasta hacer hojas de cálculo, procesamiento de textos y otros programas [77].

Esta placa cuenta con una arquitectura ARM y el sistema operativo oficial sobre el que se monta se llama Raspbian OS, una versión de Debian. Además, se pueden instalar otros sistemas como Windows 10 IoT, que tiene menos recursos que el Windows 10 de escritorio, u otros sistemas basados en Linux.

Una de las ventajas más importantes de este ordenador, aparte de que su tamaño es comparable al de una tarjeta de crédito, es su precio bajo. Este modelo hoy en día tiene un coste aproximado de 35 Euros.

Las aplicaciones de este dispositivo son muy numerosas. Básicamente depende de la imaginación del usuario ya que la funcionalidad de la Raspberry Pi es muy flexible. Por ejemplo: seguimiento de objetos en movimiento, reconocimiento facial, monitorización, transmisión de datos, desactivador de drones en movimiento [78].

2.4.2. Sensores

Los sensores son dispositivos diseñados para recibir información de una magnitud del exterior y transformarla en otra magnitud, generalmente eléctrica, que se pueda cuantificar y manipular [79]. Existen diferentes tipos de sensores: de posición lineal y angular, de desplazamiento y deformación, de velocidad lineal y angular, de aceleración, de presión, de caudal, de temperatura, de presencia, táctiles, de visión artificial, de proximidad, de acústico, de acidez, de luz y de captura de movimiento [80].

2.4.3. Raspbian OS

Raspbian es una distribución del sistema operativo *GNU/Linux* basado en Debian optimizado para el hardware de Raspberry Pi [81, 82]. Raspbian no está afiliado a la Fundación Raspberry Pi, de hecho, fue creado por Mike Thompson y Peter Green como un proyecto independiente.

Al ser una distribución de *GNU/Linux* las posibilidades son infinitas ya que el usuario puede descargar multitud de programas contenidos en repositorios como si se tratase de una distribución *GNU/Linux* para equipos de escritorio.

2.4.4. Python

Python es un lenguaje de programación interpretado cuyo objetivo principal es hacer que su sintaxis favorezca un código legible. Soporta orientación a objetos, programación imperativa y rasgos de programación funcional. Es considerado un lenguaje de alto nivel, aunque su biblioteca estándar está combinada lenguajes de bajo nivel como C y C++ [83, 84].

En el proyecto, se han utilizado unas librerías específicas para los diferentes sensores como “smbus”, útil para leer información de sensores formados por circuitos inter-integrados; “RPi.GPIO” para la configuración de los pines de la Raspberry Pi; “SPIDev” usado para permitir la comunicación entre la Raspberry Pi y dispositivos periféricos. También se utilizaron otras librerías generales como “numpy”, “time”, “math”.

2.4.5. Hologram

Hologram se fundó en 2013 por Ben Forgan y Pat Wilbur en Chicago (Estados Unidos de América) con la visión de desarrollar una nueva forma de telecomunicaciones para proporcionar conectividad segura a los dispositivos IoT.

Hologram es la solución a aquellos millones de dispositivos conectados que necesitan más allá de una red WiFi para funcionar. Se basa en usar las redes celulares que pertenecen a los grandes proveedores de telefonía. Hologram negocia con ellos tener permiso a su uso verificando el hardware y estableciendo una arquitectura de datos compatible con las regulaciones. De esta manera, el modelo celular tradicional está cambiando con una solución integrada en software diseñada para expandir el alcance de la próxima generación de dispositivos conectados.

En la actualidad, el sistema de Hologram permite que cualquier cliente se conecte a más del 85% del área del planeta Tierra. El cliente solamente necesita una tarjeta SIM

perteneciente a Hologram y un módulo compatible para conectar la tarjeta con el dispositivo [85].

Además, existe una nube creada por Hologram el cual tiene las siguientes funciones: ver el tráfico de datos, enviar SMS, crear alertas o rutas para que una vez la nube reciba un mensaje determinado, actúe realizando una tarea específica (como enviar un email al usuario con una alerta, o publicar un resultado en una página web alternativa), entre otras.

2.4.6. ThingSpeak

ThingSpeak es un servicio de plataforma de análisis de *IoT* que permite añadir, visualizar y analizar flujos de datos en vivo en la nube. ThingSpeak proporciona visualizaciones instantáneas de los datos publicados por los dispositivos en ThingSpeak. Además, es posible ejecutar código MATLAB para realizar análisis y procesamiento en línea de los datos que ingresan. ThingSpeak a menudo se usa para prototipos y sistemas de prueba de concepto de *IoT* que requieren análisis [86]. Se accede a través de su API rest y para mostrar la información, se crea un canal privado (que se puede compartir otorgando permisos) en el cual se pueden ver hasta ocho gráficas con diferentes medidas.

2.4.7. Google Maps

Google Maps es un servidor de aplicaciones de mapas que fue anunciado por primera vez en 2005. Hoy en día este software gratuito pertenece a Alphabet Inc (cuya principal subsidiaria es Google).

Este software ofrece imágenes y fotografías por satélite de todo el mundo. Además, incluye imágenes de la calle con Google Street View, el tráfico en tiempo real y un guía de rutas a pie, transporte público, coche y otros modos de transporte.

2.4.8. MobaXterm

MobaXterm es un terminal mejorado para Windows con un servidor X11, un cliente *SSH* con pestañas y otras herramientas de comunicación remota como *VNC*, *telnet*, *rlogin*. En este proyecto, se utilizará MobaXterm principalmente para realizar conexiones *SSH*.

2.4.9. R y RStudio

R es un lenguaje para computación estadística y gráficos. Fue desarrollado en los laboratorios Bell y se suele usar junto con RStudio (Principal entorno de programación de R). Es un conjunto de instalaciones de software para la manipulación de datos, el cálculo y la visualización gráfica [87].

2.4.10. Draw.io

Draw.io es una tecnología perteneciente a JGraph Ltd de código abierto para crear aplicaciones de creación de diagramas, y la aplicación de diagramación gratuita de un usuario final basada en navegador más utilizada del mundo [88].

2.4.11. Comunicación

En este apartado se mostrarán los diferentes tipos de comunicación que se han usado en el proyecto.

- ❖ **Socket TCP/IP:** Proporciona un flujo de datos, generalmente de manera ordenada y fiable entre dos programas. La conexión punto a punto descrita por un socket se define de manera exclusiva por su dirección de Internet (IP), su protocolo de comunicación (Sea UDP o TCP) y el puerto, que identifica una aplicación (puede ser un puerto conocido como el SSH o 22, o uno definido por el usuario) [89].
- ❖ **SMS:** También llamado servicio de mensajes cortos o simples. Es un servicio creado junto al sistema global para las comunicaciones móviles (GSM) que está permite el envío de mensajes cortos entre teléfonos móviles [90]. El SMS sirve para teléfonos fijos y otros dispositivos de mano.
- ❖ **Correo electrónico:** también llamado *email* es un servicio de red que permite a los usuarios recibir y enviar mensajes (cartas digitales) mediante redes de comunicación electrónica [91]. Para usar este tipo de comunicación, el usuario debe disponer de una cuenta de correo electrónico del formato “usuario_ejemplo@ejemplo.com”. Cada dirección de correo electrónico se

compone de una parte local (usuario_ejemplo), el separador (@), y la parte del dominio (ejemplo.com).

- ❖ **SSH (Secure Shell)**: Es un protocolo de comunicación que fue desarrollado originalmente por Tatu Ylonen en 1995. *SSH* usa encriptación para asegurar la conexión entre un cliente y un servidor. Todas las autenticaciones del usuario, comandos, salidas y transferencias de archivos están encriptadas para evitar ataques en la red [92].

2.4.12. API REST

API REST define un conjunto de aplicaciones en los cuales los usuarios realizan peticiones y reciben respuestas utilizando el protocolo HTTP como los comandos GET y POST. Generalmente los servicios API REST son gratuitos, aunque para usar los servicios de determinadas compañías es necesario comprar el uso del servicio para tener la API REST.

Capítulo 3

Análisis y diseño

En este capítulo, se explicará en un caso ideal cómo debería ser el sistema completo. Para ello, se abordarán los diferentes requisitos del sistema para su posterior reflexión sobre los casos de uso. De esta manera, con los objetivos y los requisitos definidos, se podrá diseñar de una manera lógica la arquitectura del sistema.

3.1. Requisitos del sistema

En este apartado, se mostrarán los **requisitos funcionales**, que son todas las capacidades que describen el sistema, y los **requisitos no funcionales**, los cuales son restricciones o propiedades impuestas por sistema a desarrollar. Sin embargo, es necesario tener presente las especificaciones importantes y las que sean incompatibles con otras. Finalmente, se presentarán los casos de uso en los diferentes objetivos del proyecto.

Los requisitos del sistema se van a presentar en una tabla con la siguiente estructura [93]:

RS-XX	
Nombre	
Descripción	
Dependencias	
Importancia	
Prioridad	
Comentarios	

Tabla 3.1: Ejemplo de requisito del sistema

A continuación, se detallan los campos contenidos de la Tabla 3.1:

- ❖ **Identificador:** Es el valor que identifica de forma única cada uno de los requisitos desarrollados. La estructura del identificador mantiene la forma “RS-XX” ya que está compuesto por dos partes: “RS” indica que es un requisito del sistema y la segunda parte señala el número identificador del requisito. En la primera parte del identificador, se añade “F” si se trata de un requisito funcional y “NF” si es un requisito no funcional.
- ❖ **Nombre:** es el nombre que describe al identificador.
- ❖ **Descripción:** es la explicación detallada del requisito.
- ❖ **Dependencias:** en esta celda se indicarán qué otros requisitos son necesarios para cumplir con el requisito.
- ❖ **Importancia:** representa gradualmente la necesidad para que sea llevado a cabo el requisito. Los posibles valores son: opcional, deseable y esencial.
- ❖ **Prioridad:** representa el nivel de prioridad para que se cumpla el requisito en el sistema. Los posibles valores son: baja, media y alta.
- ❖ **Comentarios:** posibles valoraciones adicionales acerca del requisito.

3.1.1. Requisitos funcionales

RSF-01	
Nombre	Valor de la humedad en el suelo en tiempo real.
Descripción	El sistema será capaz de devolver el valor de la humedad en el suelo en tiempo real.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.2: Requisito de sistema – Requisito funcional 01

RSF-02	
Nombre	Valor de la aceleración en tiempo real.
Descripción	El sistema podrá devolver los valores de la aceleración en los ejes X, Y, Z recogidos por el giroscopio MPU6050.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.3: Requisito de sistema – Requisito funcional 02

RSF-03	
Nombre	Valor de la rotación en tiempo real.
Descripción	El sistema podrá devolver los valores de la rotación (en grados) en los ejes X e Y recogidos por el giroscopio MPU6050.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.4: Requisito de sistema – Requisito funcional 03

RSF-04	
Nombre	Detección de cuerpo térmico.
Descripción	El sistema podrá indicar si detecta un cuerpo térmico moviéndose en la zona de desarrollo.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	Valor binario (Sí o No).

Tabla 3.5: Requisito de sistema – Requisito funcional 04

RSF-05	
Nombre	Registro de los datos a lo largo del tiempo.
Descripción	El sistema podrá devolver un registro de los datos de los diferentes sensores.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.6: Requisito de sistema – Requisito funcional 05

3.1. REQUISITOS DEL SISTEMA

RSF-06	
Nombre	Interfaz web.
Descripción	Los datos del sistema podrán ser accedidos a través de una interfaz web.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.7: Requisito de sistema – Requisito funcional 06

RSF-07	
Nombre	Terminal.
Descripción	Los datos del sistema podrán ser accedidos a través de una terminal.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.8: Requisito de sistema – Requisito funcional 07

RSF-08	
Nombre	Comunicación del sistema.
Descripción	Los dispositivos del sistema serán capaces de comunicarse con el usuario.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.9: Requisito de sistema – Requisito funcional 08

RSF-09	
Nombre	Activación o desactivación del sistema de seguridad.
Descripción	El usuario podrá activar o desactivar el sistema cuando lo requiera.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.10: Requisito de sistema – Requisito funcional 09

RSF-10	
Nombre	Activación o desactivación del motor de gasoil.
Descripción	El sistema podrá activar o desactivar automáticamente el motor de gasoil en función de los parámetros obtenidos por los sensores.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	El motor de gasoil es el encargado de dar energía a la bomba de riego subterránea para sacar agua a la superficie.

Tabla 3.11: Requisito de sistema – Requisito funcional 10

RSF-11	
Nombre	Comunicación automática vía correo electrónico.
Descripción	Desde la Raspberry Pi, se enviarán alertas al usuario a su correo electrónico.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.12: Requisito de sistema – Requisito funcional 11

RSF-12	
Nombre	Comunicación automática vía SMS.
Descripción	Desde la Raspberry Pi, se enviarán alertas al usuario a su teléfono móvil.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.13: Requisito de sistema – Requisito funcional 12

3.1.2. Requisitos no funcionales

RSNF-01	
Nombre	Lenguaje de la aplicación del <i>Gateway</i> .
Descripción	El Gateway alojado en la Raspberry Pi estará desarrollado en Python.
Dependencias	****
Importancia	Deseable.
Prioridad	Alta.
Comentarios	

Tabla 3.14: Requisito de sistema – Requisito no funcional 01

RSNF-02	
Nombre	Lenguaje de la aplicación del cliente.
Descripción	El cliente estará desarrollado en Python.
Dependencias	****
Importancia	Deseable.
Prioridad	Alta.
Comentarios	

Tabla 3.15: Requisito de sistema – Requisito no funcional 02

RSNF-03	
Nombre	Sistema operativo del <i>Gateway</i> .
Descripción	El Gateway alojado en la Raspberry Pi tendrá como sistema operativo Raspbian.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta
Comentarios	

Tabla 3.16: Requisito de sistema – Requisito no funcional 03

RSNF-04	
Nombre	Datos de la humedad
Descripción	Los datos de la humedad se alojarán en la interfaz web.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.17: Requisito de sistema – Requisito no funcional 04

RSNF-05	
Nombre	Comunicación con la base de datos (interfaz web).
Descripción	La comunicación entre la Raspberry Pi y la interfaz web será posible gracias al uso de la red GSM.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.18: Requisito de sistema – Requisito no funcional 05

RSNF-06	
Nombre	Creación de túnel entre el usuario y la Raspberry Pi
Descripción	Previa a la conexión entre el usuario y la Raspberry Pi, se creará un túnel de conexión entre ambos para facilitar el sistema.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	Así podrá funcionar la comunicación vía terminal.

Tabla 3.19: Requisito de sistema – Requisito no funcional 06

RSNF-07	
Nombre	Conexión SSH entre la Raspberry Pi y el usuario
Descripción	Además de conocer los datos en la interfaz web, también el usuario podrá monitorear el sistema pidiendo instrucciones a la Raspberry Pi vía terminal a través del puerto 12345.
Dependencias	RSNF-09
Importancia	Esencial.
Prioridad	Alta
Comentarios	De esta manera, podrá ejecutar más funciones que ver los resultados en la interfaz web.

Tabla 3.20: Requisito de sistema – Requisito no funcional 07

RSNF-08	
Nombre	Descarga de la base de datos.
Descripción	El usuario podrá descargarse los datos almacenados en la interfaz web en los formatos JSON, XML o CSV.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.21: Requisito de sistema – Requisito no funcional 08

RSNF-09	
Nombre	Uso fácil del sistema.
Descripción	El usuario podrá obtener la información de cualquiera de las aplicaciones realizando menos de 5 pasos.
Dependencias	****
Importancia	Opcional.
Prioridad	Media.
Comentarios	

Tabla 3.22: Requisito de sistema – Requisito no funcional 09

RSNF-10	
Nombre	Durabilidad del sistema.
Descripción	El sistema tendrá una batería que alargará la vida del sistema y ésta se cargará al mismo tiempo cuando el motor de gasoil está activo.
Dependencias	****
Importancia	Esencial.
Prioridad	Alta.
Comentarios	

Tabla 3.23: Requisito de sistema – Requisito no funcional 10

3.1.3. Casos de uso

Un caso de uso describe una acción o actividad en el cual indica cómo se utilizará el sistema a desarrollar por sus futuros usuarios para conseguir un objetivo específico [94]. En este apartado, se muestran la especificación de los actores, los diagramas de los propios casos de uso y sus correspondientes descripciones.

3.1.3.1. Actores

Los actores o usuarios, aquellos que interactúan en el sistema, que se van a identificar en el sistema son los siguientes:

- **Usuario administrador:** Usuario que está autenticado en el sistema y tiene capacidades de administrador.
- **Usuario cliente:** Usuario normal que recibe los datos.

3.1.3.2. Diagramas de los casos de uso

A continuación, se presentarán las imágenes que representan las posibles operaciones que puede realizar el usuario en el sistema en cada uno de los tres escenarios posibles.

En el primer escenario, se representa como se accede a la información a través de la interfaz web. Cualquier usuario necesita iniciar sesión, pero solo el administrador puede conceder permisos para que los clientes puedan ver la información. Como se puede ver en la figura 3.1, las flechas verdes corresponden a las tareas disponibles del administrador, las azules corresponden a las del cliente y las naranjas a ambos.



Figura 3.1: Esquema del escenario 1 – Interfaz Web

En el segundo escenario, el administrador puede acceder a los datos a través de la terminal. Solo él puede acceder mediante un inicio de sesión. Además, puede apagar o encender el dispositivo y parar o iniciar procesos.

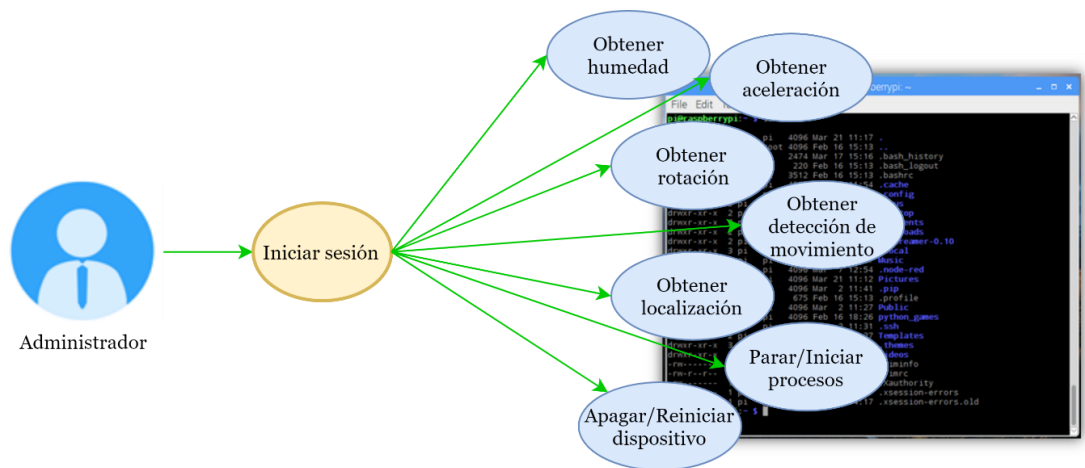


Figura 3.2: Esquema del escenario 2 – Terminal

El tercer escenario representa cómo interactúan el cliente y el administrador con el servidor. El cliente deberá tener red móvil para recibir SMS y acceso a Internet para recibir correos electrónicos, mientras que el administrador necesitará acceso a Internet para acceder al servidor.

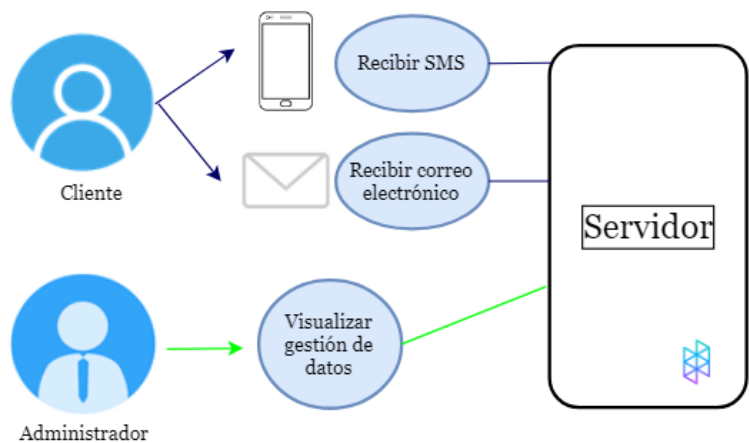


Figura 3.3: Esquema del escenario 3 – Servidor

3.1.3.3. Descripción de los casos de uso

Los casos de uso se representarán en una tabla con la siguiente estructura:

CdU-EX-XX	
Nombre	
Actores	
Descripción	
Precondiciones	

Operación	
Postcondiciones	
Comentarios	

Tabla 3.24: Ejemplo de caso de uso

A continuación, se detallan los contenidos de la Tabla 3.26:

- ❖ **Identificador:** Es el valor que identifica de forma única cada uno de los casos de uso desarrollados. La estructura del identificador mantiene la forma “CdU-EX-XX” ya que se compone de tres partes: “CdU” indica que es un caso de uso, la segunda parte señala el número de escenario descrito y la tercera parte indica el número de caso de uso.
- ❖ **Nombre:** es el nombre que describe al identificador.
- ❖ **Actores:** indica los usuarios que participan.
- ❖ **Descripción:** es la explicación detallada del caso de uso.
- ❖ **Precondiciones:** indica los requisitos que se tienen que dar para cumplir con el caso de uso.
- ❖ **Operación:** indica los pasos que debe realizar el usuario para cumplir con el caso de uso.
- ❖ **Postcondiciones:** consecuencias del sistema en el caso de uso.
- ❖ **Comentarios:** posibles valoraciones adicionales acerca del requisito.

A) Caso de uso para el escenario 1 (Interfaz Web)

CdU-E1-01	
Nombre	Inicio de sesión en la página web
Actores	Cliente y Administrador
Descripción	Para visualizar la información del sensor de humedad en la forma gráfica o descargar la información, es necesario introducir las credenciales.
Precondiciones	El usuario debe tener conexión a Internet.
Operación	1. El usuario accede a la página web 2. Inserción de sus credenciales
Postcondiciones	El usuario se encontrará en la página principal de canales y podrá seleccionar el canal de la humedad del suelo.
Comentarios	Para acceder al canal, es necesario tener permisos del Administrador.

Tabla 3.25: Escenario 1 – Caso de uso 1

CdU-E1-02	
Nombre	Visualización de valores de la humedad del suelo
Actores	Cliente y Administrador.
Descripción	El usuario visualiza el valor actual de la humedad del suelo y en un registro a lo largo del tiempo en una gráfica.
Precondiciones	El nodo que recoge los valores de la humedad debe estar encendido, junto con la Raspberry Pi como Gateway. El servidor debe funcionar para mandar la información a la interfaz web. El usuario debe tener conexión a Internet y tener permisos por parte del Administrador para acceder al canal.
Operación	<ol style="list-style-type: none"> 1. Se inicia sesión en la página web 2. Se pulsa el botón de canales para seleccionar el canal del sensor de humedad. 3. Visualización de la gráfica
Postcondiciones	La página web cambia su estado para mostrar una gráfica con el valor actual de la temperatura. Además, se muestran el resto de los valores almacenados en el registro a lo largo del tiempo.
Comentarios	

Tabla 3.26: Escenario 1 – Caso de uso 2

CdU-E1-03	
Nombre	Descarga de todos los valores de la humedad
Actores	Cliente y Administrador
Descripción	El usuario puede descargar todos los valores obtenidos en los formatos JSON, CSV o XML.
Precondiciones	El nodo que recoge los valores de la humedad debe estar encendido, junto con la Raspberry Pi como Gateway. El servidor debe funcionar para mandar la información a la interfaz web. El usuario debe tener conexión a Internet y tener permisos por parte del Administrador para acceder al canal.
Operación	<ol style="list-style-type: none"> 1. Se inicia sesión en la página web. 2. Se pulsa el botón de canales para seleccionar el canal del sensor de humedad. 3. Se pulsa el botón de exportar información para seleccionar el formato deseado y proceder a la descarga.
Postcondiciones	El usuario obtendrá un fichero en que podrá acceder a los datos del sensor de humedad desde el principio hasta ese momento.
Comentarios	Como característica de la página web, se almacenan los datos en el canal, aunque la Raspberry Pi se reinicie o mande nueva información, se mantendrá toda la información previa.

Tabla 3.27: Escenario 1 – Caso de uso 3

CdU-E1-04	
Nombre	Concesión de permisos a clientes.
Actores	Administrador.
Descripción	El administrador puede conceder permisos a los clientes para ver o descargar la información de la humedad del suelo ya que esta información no es pública.
Precondiciones	El administrador debe mantener conexión a Internet.
Operación	<ol style="list-style-type: none"> 1. Iniciar sesión en la página web. 2. Pulsar “sharing” o compartir. 3. Seleccionar “Share channel view only with the following users” o compartir solo con los siguientes usuarios. 4. Introducir cuenta del cliente.
Postcondiciones	El administrador volverá a la página principal. El cliente tendrá los permisos para acceder.
Comentarios	

Tabla 3.28: Escenario 1 – Caso de uso 4

B) Casos de uso para el escenario 2 (Terminal)

CdU-E2-01	
Nombre	Inicio de sesión en la terminal.
Actores	Administrador.
Descripción	Es necesario introducir las credenciales del usuario para acceder al sistema completo.
Precondiciones	Tanto el usuario como la Raspberry Pi necesitan conexión a Internet.
Operación	1. Introducir usuario y contraseña
Postcondiciones	El administrador tendrá acceso total al sistema.
Comentarios	

Tabla 3.29: Escenario 2 – Caso de uso 1

CdU-E2-02	
Nombre	Obtener información sobre la humedad.
Actores	Administrador.
Descripción	El usuario puede seleccionar ver la humedad actual en ese momento.
Precondiciones	El nodo sensor debe estar funcionando y tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet. Se necesita autenticar al usuario para acceder al sistema.
Operación	<ol style="list-style-type: none"> 1. Iniciar sesión en el programa 2. Pedir información de la humedad

3.1. REQUISITOS DEL SISTEMA

Postcondiciones	El usuario obtendrá la información de la humedad en la terminal.
Comentarios	

Tabla 3.30: Escenario 2 – Caso de uso 2

CdU-E2-03	
Nombre	Obtener información sobre la aceleración.
Actores	Administrador.
Descripción	El usuario puede seleccionar ver la aceleración actual en los tres ejes del nodo sensor en ese momento.
Precondiciones	El nodo sensor debe estar funcionando y tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet. Se necesita autenticar al usuario para acceder al sistema.
Operación	<ol style="list-style-type: none">1. Iniciar sesión en el programa2. Pedir información de la aceleración
Postcondiciones	El usuario obtendrá la información de la aceleración en la terminal.
Comentarios	

Tabla 3.31: Escenario 2 – Caso de uso 3

CdU-E2-04	
Nombre	Obtener información sobre la rotación.
Actores	Administrador.
Descripción	El usuario puede seleccionar ver la rotación actual en los ejes X e Y del nodo sensor en ese momento.
Precondiciones	El nodo sensor debe funcionar y la Raspberry y el usuario tendrán conexión a Internet. Iniciar sesión para acceder al sistema.
Operación	<ol style="list-style-type: none">1. Iniciar sesión en el programa2. Pedir información de la rotación
Postcondiciones	El usuario obtendrá la información de la rotación en la terminal.
Comentarios	

Tabla 3.32: Escenario 2 – Caso de uso 4

CdU-E2-05	
Nombre	Obtener información sobre cuerpos térmicos en movimiento.
Actores	Administrador.
Descripción	El usuario puede ver si se han detectado cuerpos térmicos en movimiento por la zona y saber cuántas veces.
Precondiciones	El nodo sensor debe estar funcionando y tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet. Se necesita autenticar al usuario para acceder al sistema.
Operación	<ol style="list-style-type: none">1. Iniciar sesión en el programa

	2. Pedir información sobre los cuerpos en movimiento
Postcondiciones	El usuario obtendrá la información del número de cuerpos en la terminal.
Comentarios	

Tabla 3.33: Escenario 2 – Caso de uso 5

CdU-E2-06	
Nombre	Obtener información de la localización del dispositivo.
Actores	Administrador.
Descripción	El administrador podrá ver donde se encuentra el dispositivo en todo momento recibiendo las coordenadas de éste.
Precondiciones	Tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet. Se necesita autenticar al usuario para acceder al sistema.
Operación	1. Iniciar sesión en el programa 2. Pedir información de la localización
Postcondiciones	El usuario podrá ver en la terminal los valores de la latitud y la longitud.
Comentarios	

Tabla 3.34: Escenario 2 – Caso de uso 6

CdU-E2-07	
Nombre	Apagar o reiniciar el dispositivo.
Actores	Administrador.
Descripción	El administrador podrá apagar o reiniciar el dispositivo para labores de mantenimiento o mejora del sistema.
Precondiciones	Tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet. Se necesita autenticar al usuario para acceder al sistema.
Operación	1. Iniciar sesión en el programa 2. Escribir “sudo shutdown -h now” en caso de apagar o “sudo reboot” en caso de reiniciar.
Postcondiciones	El usuario podrá ver si ha reiniciado o se ha apagado.
Comentarios	

Tabla 3.35: Escenario 2 – Caso de uso 7

CdU-E2-08	
Nombre	Parar y/o iniciar procesos.
Actores	Administrador.
Descripción	El administrador podrá parar y/o iniciar procesos por temas de mantenimiento o mejora en el sistema.
Precondiciones	Tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet. Se necesita autenticar al usuario para acceder al sistema.

Operación	<ol style="list-style-type: none"> 1. Iniciar sesión en el programa 2. Buscar el identificador del proceso concreto 3. Escribir “sudo kill <Identificador>”
Postcondiciones	El proceso se habrá parado o iniciado.
Comentarios	

Tabla 3.36: Escenario 2 – Caso de uso 8

C) Casos de uso para el escenario 3 (Servidor)

CdU-E3-01	
Nombre	Recibir alertas SMS.
Actores	Cliente.
Descripción	El cliente recibirá una alerta del servidor en forma de SMS si se cumplen los requisitos. Además, recibirá en el mismo mensaje las coordenadas del nodo de localización.
Precondiciones	Los nodos sensores deben estar funcionando y la Raspberry Pi debe mantener la conexión a Internet y el servidor debe tener saldo para poder enviar un SMS al cliente.
Operación	El usuario debe esperar al mensaje.
Postcondiciones	El usuario podrá ver un nuevo mensaje en su teléfono móvil y acceder a la localización.
Comentarios	Las alertas pueden deberse por detección de cuerpo térmico o movimiento indebido del motor de gasolina.

Tabla 3.37: Escenario 3 – Caso de uso 1

CdU-E3-02	
Nombre	Recibir alertas vía correo electrónico.
Actores	Cliente.
Descripción	El cliente recibirá una alerta del servidor en forma de correo electrónico si se cumplen los requisitos. Además, recibirá en el mismo mensaje las coordenadas del nodo de localización.
Precondiciones	Los nodos sensores deben estar funcionando y tanto la Raspberry Pi como el usuario deben mantener la conexión a Internet.
Operación	<ol style="list-style-type: none"> 1. Iniciar sesión en la cuenta de correo electrónico 2. Abrir “bandeja de entrada”
Postcondiciones	El usuario podrá ver un nuevo mensaje en su bandeja de entrada.
Comentarios	Las alertas pueden deberse por detección de cuerpo térmico o movimiento indebido del motor de gasolina.

Tabla 3.38: Escenario 3 – Caso de uso 2

CdU-E3-03	
Nombre	Gestión de datos.
Actores	Administrador.
Descripción	El administrador podrá ver el consumo de datos móviles de la Raspberry Pi en una gráfica y en un formato numérico.
Precondiciones	El usuario debe tener conexión a Internet para acceder al consumo de datos móviles.
Operación	<ol style="list-style-type: none"> 1. Iniciar sesión en la página del servidor 2. Pulsar “Dashboard” o panel de control. 3. Hacer clic en la tarjeta SIM que se está usando en la Raspberry Pi. 4. Hacer clic en “Usage” o Uso
Postcondiciones	El usuario podrá visualizar el consumo de datos totales de manera numérica y en una gráfica.
Comentarios	

Tabla 3.39: Escenario 3 – Caso de uso 3

3.2. Arquitectura del sistema

A partir del análisis realizado en el apartado anterior, se saben qué objetivos y requisitos se deben cumplir. A continuación, se describe en un modo lógico cómo debería ser la arquitectura del sistema:

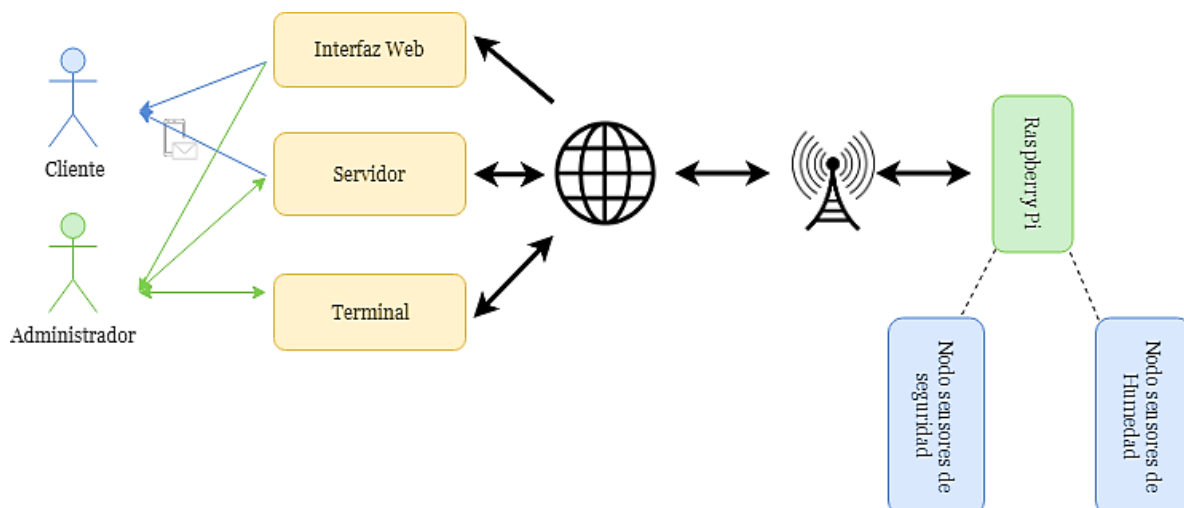


Figura 3.4: Arquitectura ideal del sistema

Como se puede ver en la Figura 3.4, se pueden distinguir varias partes. En primer lugar, en la parte de la derecha se encuentra la Raspberry Pi junto con los nodos que recogen la

información del entorno. En segundo lugar, en la parte central, la Raspberry Pi envía la información a Internet a través de las torres GSM. En último lugar, una vez llega a Internet, la información se deposita en los diferentes escenarios en los cuales el administrador y el cliente tienen diferentes roles.

Como ya se explicó en el punto anterior, se listan tres escenarios: la Interfaz Web, en los cuales el cliente y el administrador pueden visualizar la información de la humedad en un formato gráfico y descargar la misma en diferentes formatos; el servidor, en que los usuarios tienen diferentes operaciones y la terminal, en el que solo el administrador puede entrar para comprobar los diferentes parámetros del sistema a larga distancia.

Además, se puede ver en el gráfico flechas unidireccionales y bidireccionales. Esto significa que la información podrá viajar en un solo sentido (unidireccional) como la flecha de la interfaz web (recepción de información y no envío) o en ambos (bidireccional) como el administrador y la terminal ya que cuando el administrador pide información, la recibe. Las flechas también están coloreadas en función del usuario: verde para el administrador y azul para el cliente.

Capítulo 4

Implementación

En este capítulo se explicará paso a paso cómo se ha desarrollado todo el proyecto de acuerdo con los objetivos presentados en el capítulo 1 y los requisitos del sistema, además de la arquitectura del sistema. Se decidió crear dos programas: uno para controlar automáticamente el riego y mandar información a la interfaz web y otro para monitorear el sistema de seguridad. Por otro lado, se mostrará cómo el administrador se comunicará con el sistema en un tercer modo de comunicación. Una vez todo junto, se simulará y se hará una prueba en la realidad.

4.1. Contexto

La vid necesita más o menos agua dependiendo de la estación del año en la que se encuentre. Como podrá observar en el anexo 2, además, se necesita recoger la cantidad óptima de agua subterránea para tener una recolección eficiente de la uva y para evitar agotar el acuífero. Para ello, también hay que tener en cuenta la lluvia útil en las diferentes estaciones.

Por otro lado, también es necesario observar el estado del motor de gasoil y alertar al usuario cuándo se está produciendo un robo del motor ya que supone una inversión económica considerable para el agricultor.

4.2. Programa de riego – riego.py

En este programa, se tienen que respetar los siguientes objetivos: obtener el valor de humedad en el momento, enviarlo a la plataforma web para su visualización en una gráfica, encender y apagar el motor de gasoil y hacer una copia de los parámetros utilizados en el programa en un archivo log. Esta información se resume en el siguiente diagrama de flujo:

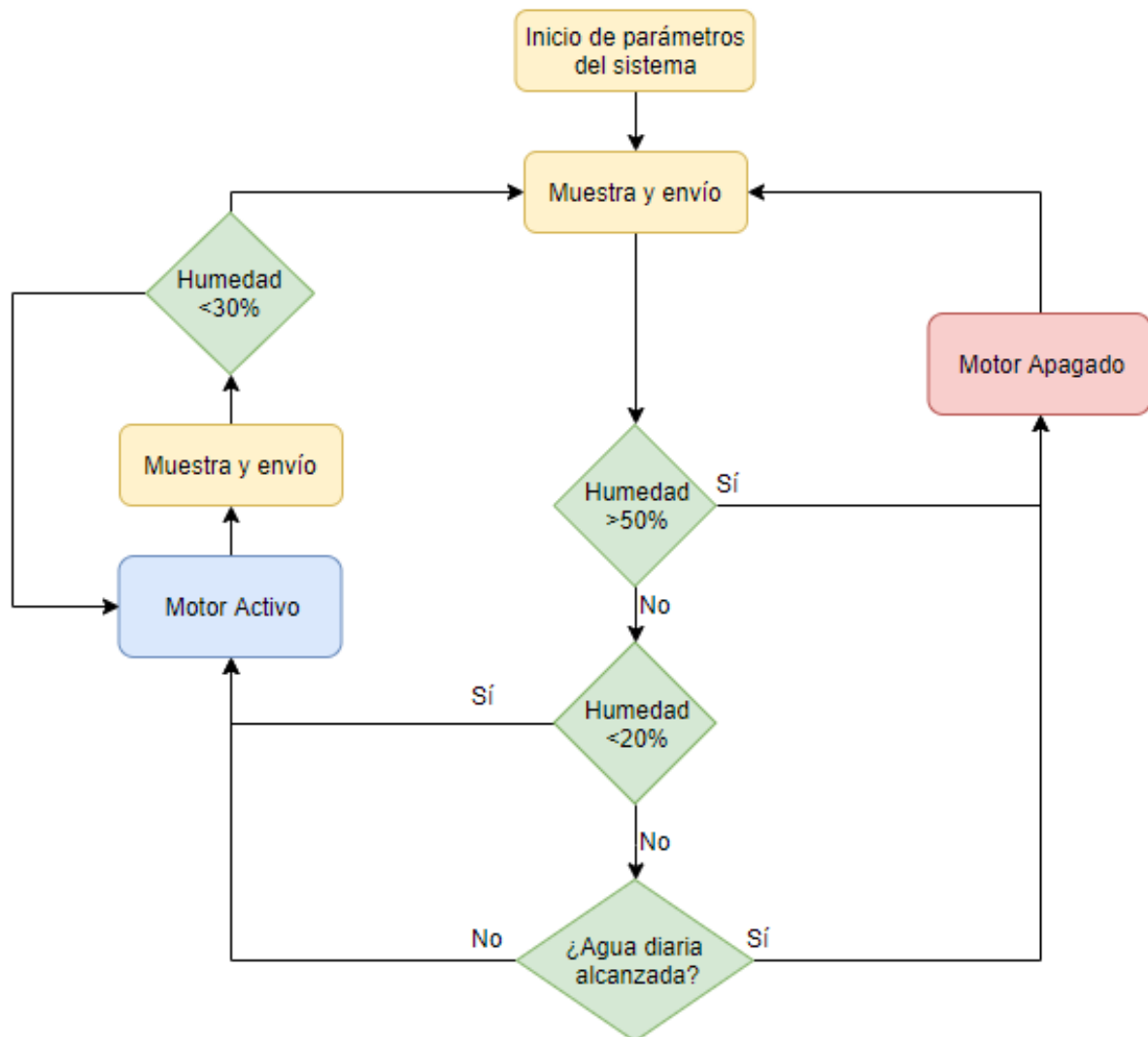


Figura 4.1: Diagrama de flujo – riego.py

Junto con **riego.py** existe otro archivo auxiliar llamado **mcp3008.py** que se encarga de recoger la información del sensor de humedad gracias al convertidor analógico – digital.

4.2.1. Análisis de la humedad

Habrán dos requerimientos fundamentales a la hora de activar el motor o no: La humedad del suelo y la cantidad de agua mínima necesaria. Ciertamente, que haya agua implica que habrá humedad, pero en las épocas de floración, envero y maduración se necesita más agua para que vaya directamente al fruto y este crezca, además de un nivel de humedad mínimo asegurará que la planta no se seque (más información en el anexo 2).

En primer lugar, si los niveles de humedad crecen considerablemente (más de un 50%), entonces no es necesario aplicar el riego, ya que tampoco es recomendable y con ese nivel la planta puede desarrollar el fruto correctamente. Ahora bien, en segundo lugar, si el nivel de humedad es inferior al 20%, entonces no se tiene en cuenta el nivel de agua alcanzada hasta que se alcance el 30% de humedad (De esta manera el nivel no está justo en el límite inferior).

Hay una tercera condición: el nivel de humedad que se encuentre entre el 20% y el 50%. En este caso, no se regará más si se ha alcanzado el agua disponible diaria y de lo contrario, regará hasta que llegue al límite del 50% o se haya agotado el agua destinada para ese día.

Al ser una implementación, se programará un led azul que actúa de motor de gasoil. Se encenderá para simular que el motor se enciende y se apagará en caso contrario.

4.2.2. Cálculo del agua alcanzada

De una manera orientativa, para calcular la cantidad de agua que se está regando por metro cuadrado o agua alcanzada, se utilizará la información que tiene el motor de gasoil que se presentó en la figura 1.1 [2]. Este motor de gasoil es capaz de alimentar a una bomba que extrae 8300 litros por hectárea y hora. Eso quiere decir:

$$\begin{aligned} \text{Rítm}_o_{\text{Extracción}} &= 8300 \frac{\text{litros}}{\text{hectárea} \cdot \text{hora}} \cdot \frac{\text{hectárea}}{10000 \text{ m}^2} \\ &= 0.83 \frac{\text{litros}}{\text{m}^2 \cdot \text{hora}} \cdot \frac{1 \text{ hora}}{3600 \text{ segundos}} = 2.3055 \cdot 10^{-4} \frac{\text{litros}}{\text{m}^2 \cdot \text{segundo}} \end{aligned}$$

Si el motor funciona una hora exactamente, se tendrá 0.83 litros en un metro cuadrado. Entonces para calcular el agua alcanzada, se aplicaría el siguiente cálculo:

$$\text{Agua alcanzada} = \text{Rítm}_o_{\text{Extracción}} \cdot \text{tiempo}_{\text{Activo}}$$

En riego.py, el tiempo activo crece cuando se cumplen las condiciones de extracción. Como se leen datos cada hora, cada ciclo del “while” sumará 3600 segundos al tiempo parado o al tiempo activo. En el programa se utilizará segundos como medida de tiempo, por lo tanto, ciclo_1 será igual a 3600 y el ritmo a $2.3055 \cdot 10^{-4}$.

```
while True:
    if first== 1:
        first = 0
    else:
        if led == 1:
            tiempo_motor= tiempo_motor+ciclo_1
        else:
            tiempo_parada=tiempo_parada+ciclo_1
```

Figura 4.2: Tiempo motor (activo) y tiempo parada cada hora

4.2.3. Cálculo de agua diaria disponible

De acuerdo con los cálculos realizados en el anexo 2, es necesario tener en cuenta las precipitaciones del año ya que cuanto más llueva en la zona, menos agua será necesaria para el riego. De un modo orientativo, se calculó la lluvia útil que caía en Corral de Almaguer en el año 2017 y se restó al agua necesaria en la vid por metro cuadrado. De esta manera se tenía el agua que era necesario en un mes, por lo tanto, para conseguir el agua necesaria diaria, se debía dividir el agua mensual por el número de días en cada mes (Resultado en la Tabla A.2.4).

4.2.4. Envío de datos a *ThingSpeak*

A medida que el sensor recoja los datos de la humedad, el programa los irá enviando automáticamente a la página de *ThingSpeak*. Para ello, en riego.py, primero, se conectará

a la plataforma para que se permita el envío de datos. Se escribirá la clave de escritura API y se enlazará con el URL de la página web:

```
#ThingSpeak setup
myAPI = "1C6E8L5UY683QB39" #Usar la API Key de escritura de ThingSpeak
baseURL = 'https://api.thingspeak.com/update?api_key=%s' % myAPI
```

Figura 4.3: Configuración de los parámetros para acceder a ThingSpeak

Para enviar la información, se necesitará realizar una POST *request* de la siguiente manera:

```
#ThingSpeak.setup
myAPI = "1C6E8L5UY683QB39" #Usar la API Key de escritura de ThingSpeak
baseURL = 'https://api.thingspeak.com/update?api_key=%s' % myAPI

def send_message_to_thingspeak(mensaje_):
    try:
        f = urllib2.urlopen(baseURL+"&field1=%s"%(mensaje_))
        f.close()
    except:
        fl.write('Conexion fallida, intentando de nuevo...')
```

Figura 4.4: Función para enviar mensaje a la plataforma

Como se puede ver en la función “urlopen”, se necesita la dirección URL de la figura 4.2 junto con la clave de escritura. Además, para mandar la información es necesario que se elija qué gráfica mostrará los datos. En *ThingSpeak*, se pueden crear hasta ocho gráficas, pero como se está mostrando solo la humedad, se envía a la primera gráfica llamada “field1”.

Channel Settings

Percentage complete 30%

Channel ID

Name

Description

Field 1 ☒

Field 2 ☐

Field 3 ☐

Field 4 ☐

Field 5 ☐

Field 6 ☐

Field 7 ☐

Field 8 ☐

Figura 4.5: Configuración del canal en la página web

Una vez el programa está funcionando y los datos vayan llegando a la plataforma, se podrá ver un resultado como en la siguiente figura:

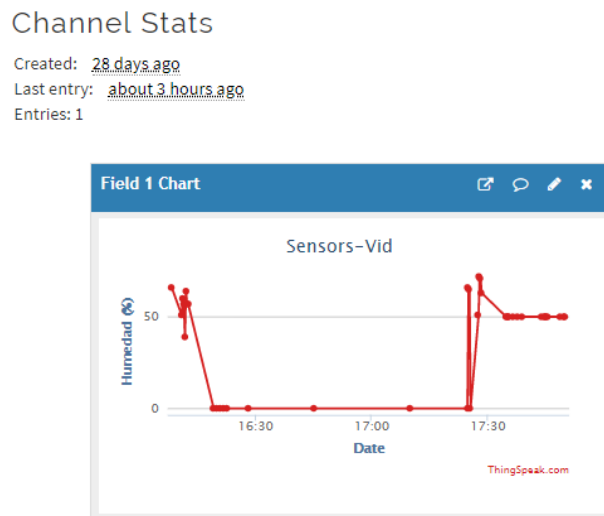


Figura 4.6: Gráfica de la humedad en tiempo real

A su vez, si el usuario estuviese interesado en conseguir la información “en crudo”, puede hacerlo descargando el fichero en los formatos JSON, XML o CSV. Para ello, debe hacer clic en exportar los datos recientes (*export recent data*) en la página principal del canal seleccionado.

4.2.5. Creación del fichero “log_riego.txt”

En el caso de que ocurriese algún problema con la transmisión de los datos o se deseara ver qué movimientos ocurrieron en el pasado, existe un archivo llamado “**log_riego.txt**” que se almacenará en la Raspberry Pi.

En este archivo, la información se insertará en el formato: <fecha> <agua utilizada> <Humedad> <tiempo motor> <tiempo parado>. El administrador podrá acceder al fichero utilizando la terminal (punto 4.4).

```
29-08-2019 17:25:10 Agua utilizada: 0.000276666 Litros - Humedad: 67% - Tiempo Motor: 12 s - Tiempo Parado: 8 s
29-08-2019 17:25:13 Agua utilizada: 0.000276666 Litros - Humedad: 66% - Tiempo Motor: 12 s - Tiempo Parado: 10 s
29-08-2019 17:25:16 Agua utilizada: 0.000276666 Litros - Humedad: 65% - Tiempo Motor: 12 s - Tiempo Parado: 12 s
29-08-2019 17:25:18 Agua utilizada: 0.000276666 Litros - Humedad: 64% - Tiempo Motor: 12 s - Tiempo Parado: 14 s
29-08-2019 17:25:21 Agua utilizada: 0.000276666 Litros - Humedad: 63% - Tiempo Motor: 12 s - Tiempo Parado: 16 s
29-08-2019 17:25:24 Agua utilizada: 0.000276666 Litros - Humedad: 61% - Tiempo Motor: 12 s - Tiempo Parado: 18 s
29-08-2019 17:25:27 Agua utilizada: 0.000276666 Litros - Humedad: 0% - Tiempo Motor: 12 s - Tiempo Parado: 20 s
29-08-2019 17:25:29 Agua utilizada: 0.000322777 Litros - Humedad: 0% - Tiempo Motor: 14 s - Tiempo Parado: 20 s
29-08-2019 17:25:32 Agua utilizada: 0.000368888 Litros - Humedad: 0% - Tiempo Motor: 16 s - Tiempo Parado: 20 s
29-08-2019 17:25:34 Agua utilizada: 0.000414999 Litros - Humedad: 0% - Tiempo Motor: 18 s - Tiempo Parado: 20 s
```

Figura 4.7: Ejemplo de “log_riego.txt”

4.3. Detección de robo

Se valorará cuándo se está produciendo un robo o no. Para ello, en el programa principal, **seguridad.py**, se comprobarán los siguientes parámetros:

- Existencia de un cuerpo térmico (persona) en movimiento en la zona.
- Movimiento del motor (velocidad y rotación)

En el caso de que se detectase una actividad sospechosa, se realizará un envío de una alerta al cliente explicando cómo se ha detectado y acto seguido, se insertará un enlace a Google Maps con la localización en latitud y longitud obtenido por el módem *GSM*. Esta información se puede visualizar en el siguiente diagrama de flujo:

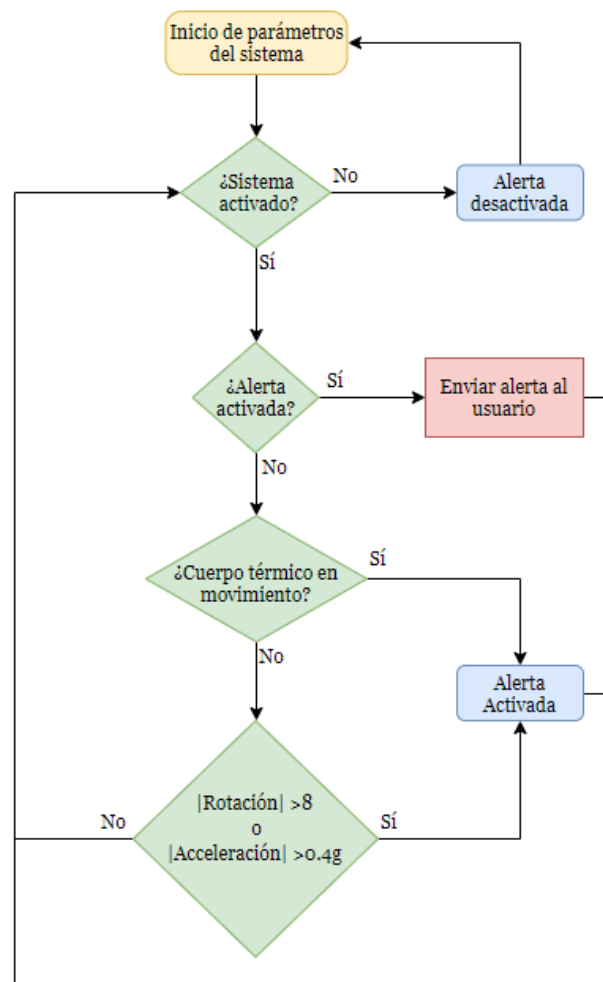


Figura 4.8: Esquema de funcionamiento – `seguridad.py`

4.3.1. Sistema de activación y desactivación del programa

Una de las cosas más importantes que debe tener el sistema es la activación y la desactivación de la seguridad ya que el cliente debe ir a recargar el depósito de gasoil para que la máquina siga funcionando. Si no existiese una forma de paralizar la seguridad mientras el agricultor está llenando el depósito, recibiría muchos mensajes de alerta. Por otro lado, también es necesario que se inicie sesión en el sistema de seguridad para poder enviar los mensajes de activación y desactivación ya que cualquiera podría enviar mensajes al dispositivo.

Hologram ofrece dos opciones para enviar SMS al dispositivo: que los envíe el administrador o que los envíe el cliente. En el primer caso, el cliente tendría que especificar al administrador cuándo debería enviar un SMS para parar y otro para activar el programa de seguridad. Esto es bastante laborioso, pero el administrador en el servicio de la nube de Hologram puede enviar mensajes gratuitos al dispositivo. En el segundo caso, para que el cliente los pueda enviar sin tener que llamar al administrador, tendría que adquirir un número de teléfono para la tarjeta SIM Hologram. En este caso, el administrador contrataría un número de teléfono por 1 dólar al mes y el cliente podría enviar todos los SMS a este número¹.

Continuando con la implementación del sistema de activación, se crea un programa llamado `dueno.py` que contiene cuatro funciones auxiliares para su uso en `seguridad.py`. La primera función se llama `randomString(stringLength)` que consiste en crear una palabra de manera aleatoria y usa la segunda función, `envio_clave(cloud)` para enviarla al número de teléfono del cliente.

Una vez el cliente reciba la notificación de que el sistema se ha encendido, tiene que enviar un SMS con la clave que él mismo reciba para activarlo. La tercera función, `activar_sistema(cloud)` estará revisando si hay SMS en la bandeja de entrada y si los hay, comparará la clave con el SMS recibido. Si no es correcto, enviará un SMS de vuelta diciendo "clave incorrecta" y si lo es, activará el sistema de seguridad y notificará al cliente con otro SMS.

¹ Es posible conectar otros tipos de tarjetas SIM, incluso aquellos que soporten cobertura española utilizando módems, módulos instalables o las tarjetas incrustadas (embedded board). Sin embargo, muchas compañías de telefonía carecen de servicios que Hologram si ofrece, como el panel de control en su página web, el servicio SpaceBridge (Véase apartado 4.4), o el encaminamiento de rutas (punto 4.3.5).

La cuarta función, “`a_d_sms (cloud)`”, desactivará y activará el sistema de seguridad cuando reciba un mensaje del cliente. Funcionará si el sistema de seguridad se ha activado al menos una vez y será útil para parar el sistema por un momento para rellenar el depósito de gasoil u hacer otras tareas cerca de la máquina. Esta función mira si hay un SMS nuevo en la memoria y proporciona una respuesta que depende del SMS:

- ❖ Si no existe SMS, la respuesta será ‘U’ de *unknown* o desconocido.
- ❖ Si existe SMS y su mensaje es ‘D’ o ‘d’, la respuesta será ‘D’ o desactivado.
- ❖ Si existe SMS y su mensaje es ‘A’, la respuesta será ‘A’ o activado.
- ❖ Si existe SMS y su mensaje no es ni ‘D’ ni ‘A’ (o en minúscula), la respuesta será ‘U’ de *unknown* o desconocido.

En el programa principal, `seguridad.py`, se ejecutarán las tres primeras funciones de `dueno.py` y después, habrá un bucle infinito que siempre llamará a la cuarta función y al resto de funciones de seguridad. El sistema está activo y está constantemente vigilando todos los parámetros cada 2 segundos. Sin embargo, cuando se desactiva, el sistema vuelve a llamar a la cuarta función cada 15 minutos por si hay un mensaje de activación.

4.3.2. Detección de cuerpos térmicos en movimiento

Para poder variación de movimiento en los cuerpos térmicos de alrededor es necesario utilizar el sensor *PIR*. Como ya se habló en el capítulo 1, el *PIR* es un sensor que detecta variaciones en las radiaciones infrarrojas.

Sin embargo, existe un problema con este sensor: puede detectar cualquier variación de radiación infrarroja, es decir, puede detectar una persona o un animal. Esto supone otro problema porque pueden ocurrir falsas alarmas.

Para ello, se tiene que aprovechar la altura en la que se coloca el sensor ya que el *PIR* establece una línea de detección con las lentes del propio sensor. Si un animal está debajo de esa línea, no será detectado. Por lo tanto, sabiendo que los animales de la zona son pequeños, la línea de detección puede estar a un metro y medio de altura.

Así pues, en `seguridad.py`, para comprobar el funcionamiento del *PIR*, se comprueba si se recibe alguna señal en el pin elegido para el sensor. En la Figura 4.9, se incluye la fecha y la hora cuando se recibe una señal positiva.

```
if GPIO.input(PIR_PIN):#Si detecta movimiento por cuerpo termico
    timex = strftime("%d-%m-%Y %H:%M:%S", gmtime()) #Creamos una cadena de texto con la hora
    timex = timex + " MOVIMIENTO DETECTADO" #La sacamos por pantalla
    print timex
```

Figura 4.9: Sentencia condicional con el sensor PIR

4.3.3. Análisis del movimiento angular y lineal del motor

Esta herramienta es un recurso adicional para la seguridad del motor. Puede ser como una segunda medida de seguridad si el *PIR* no llegase a detectar un cuerpo térmico por variación de radiaciones infrarrojas en la zona.

Para ello, el sensor que se utiliza es el MPU6050. Este sensor se colocará en una superficie plana en el motor y analizará la rotación y la aceleración de este. En condiciones normales, el motor se encuentra en una posición estática, es decir, su movimiento angular (rotación) y la velocidad lineal no varían en ningún momento. En el momento en el que el dueño u otra persona quisiera mover el motor, habría variaciones en estos parámetros.

Para ello, se han hecho pruebas para ver el funcionamiento del sensor sin ningún movimiento. En la Figura 4.10 se encuentran los resultados numéricos mientras que en la Figura 4.11 se puede ver que el sensor está sobre una superficie plana (una mesa):

```
Aceleracion
X speed:  0.019  Y Speed:  -0.014  Z speed:  0.955
X speed:  0.018  Y Speed:  -0.005  Z speed:  0.966
X speed:  0.018  Y Speed:  -0.011  Z speed:  0.963
X speed:  0.022  Y Speed:  -0.004  Z speed:  0.958
X speed:  0.02  Y Speed:  -0.011  Z speed:  0.963
X speed:  0.015  Y Speed:  -0.01  Z speed:  0.955
Rotacion
X Rotation:  -0.101  Y Rotation:  -1.766  Z Rotation:  88.231
X Rotation:  -0.641  Y Rotation:  -1.122  Z Rotation:  88.708
X Rotation:  -0.643  Y Rotation:  -1.609  Z Rotation:  88.267
X Rotation:  -0.347  Y Rotation:  -0.854  Z Rotation:  89.078
X Rotation:  -0.219  Y Rotation:  -1.211  Z Rotation:  88.77
X Rotation:  -0.399  Y Rotation:  -1.797  Z Rotation:  88.16
```

Figura 4.10: Pruebas del sensor MPU6050 (Aceleración y Rotación)

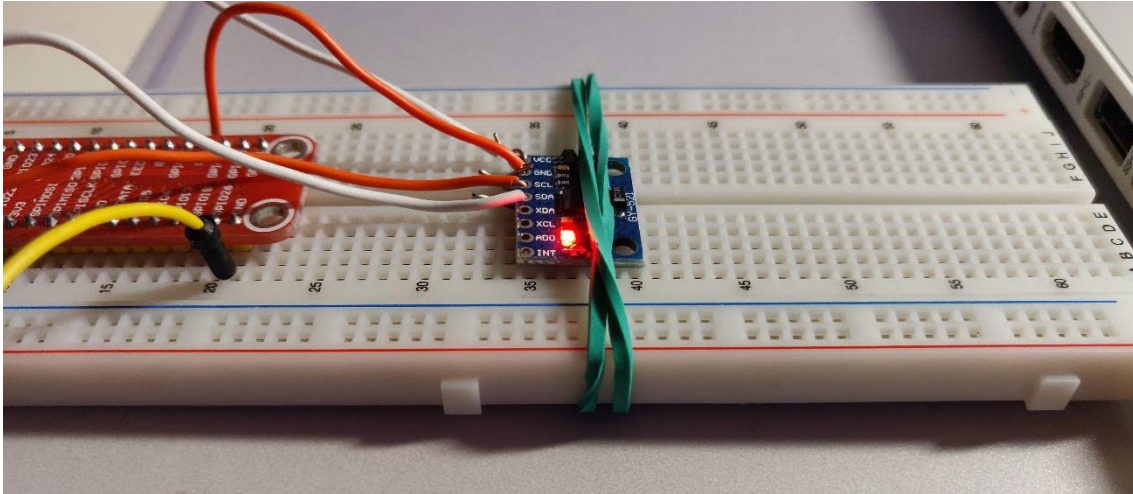


Figura 4.11: Posición del sensor MPU6050

El valor de la aceleración se mide en unidades “g”, es decir, 9.8 m/s^2 . Esta unidad corresponde con la gravedad terrestre. Sobre la superficie plana, el eje z tendrá toda la aceleración de la gravedad. Entonces, los ejes X e Y determinarán la aceleración lineal:

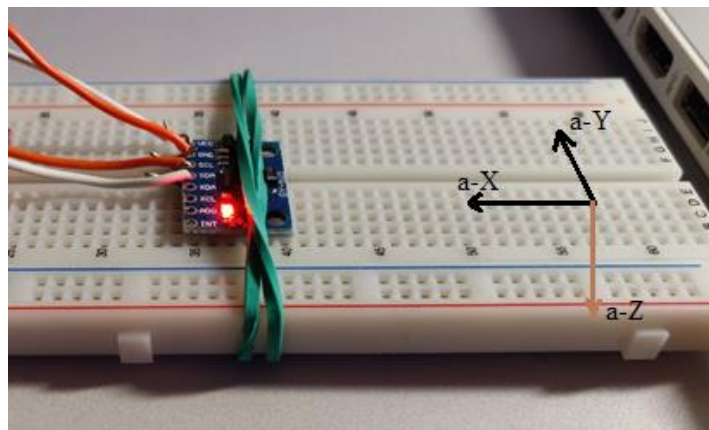


Figura 4.12: Vista de la aceleración en los diferentes ejes y el MPU6050

Como los resultados en los ejes X e Y dan valores en las centésimas en el estado inmóvil (virtualmente 0 g), el valor que determine movimiento sospechoso será mayor a 0.4g. Asumiendo que un vehículo que arrastre el motor logre 20 km/h en 3 segundos, equivaldría a acelerar a 6.66 m/s^2 o 0.68g. Por lo tanto, para agudizar más el proceso, se optará por 0.4g en los ejes X e Y como valor máximo para activar la alarma.

Por otro lado, la rotación se mide en grados. Los valores importantes son de nuevo los ejes X e Y, ya que el eje Z depende de otro giroscopio para establecer la referencia. Sin embargo, no es necesario usar el eje Z ya que con los dos primeros sería suficiente. A

continuación, en la siguiente figura se puede ver cómo aumenta la rotación en los diferentes ejes:

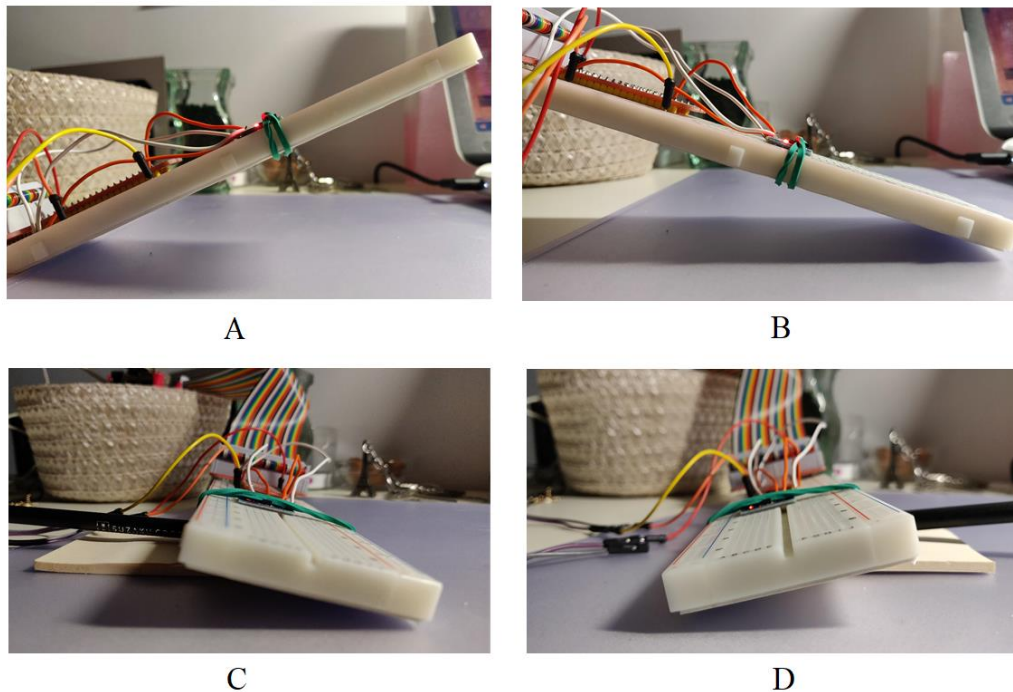


Figura 4.13: Diferentes posiciones del giroscopio

Donde:

- Imagen A: Se avanza en el eje Y negativo.
- Imagen B: Se avanza en el eje Y positivo.
- Imagen C: Se avanza en el eje X negativo.
- Imagen D: Se avanza en el eje X positivo.

Esto será útil para comprobar que está en movimiento ya que la superficie en la que se encuentra el motor es irregular y pedregoso. Por lo tanto, es fácil que el motor sobre un remolque rebote lo suficiente para que salte el sensor. Como se puede ver en los resultados de la figura 4.10, la rotación llega a variar hasta 0.6 grados en un estado inmóvil. Por lo tanto, se establecerá que el límite máximo para no alertar al usuario será de 8 grados.

```
if abs(rot_x)>8 or abs(rot_y) >8 or abs(acc_x) > 0.4 or abs(acc_y) >0.4:
    timex = strftime("%d-%m-%Y %H:%M:%S", gmtime()) #Creamos una cadena de texto con la hora
    timex = timex + " Sospecha de Robo" #La sacamos por pantalla
    print timex
```

Figura 4.14: Sentencia condicional con sensor MPU6050

4.3.4. Localización del dispositivo

En el caso de que salten las alarmas (sea por el *PIR* o el movimiento angular y lineal) se procedería a enviar la localización del motor cada 3 minutos.

Para ello, es necesario usar el módem *Hologram Nova*. Este dispositivo está conectado a uno de los USB de la Raspberry Pi:

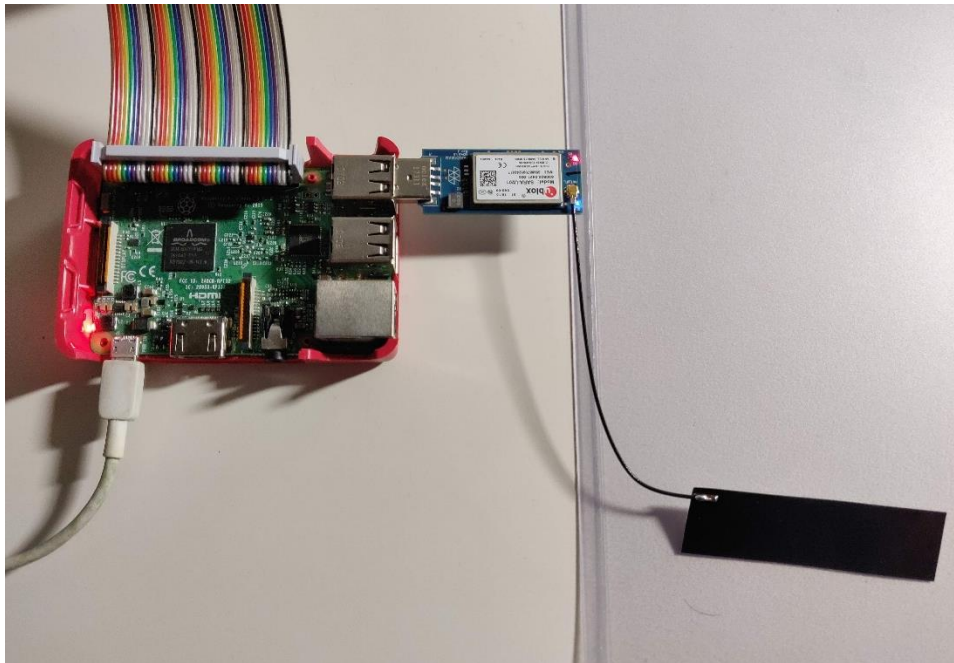


Figura 4.15: Raspberry Pi conectado al módem Hologram Nova

De esta forma, en el código se llamaría a la función `get_coordinates()` que conseguiría los parámetros del módem: “`network.location.latitude`” y “`network.location.longitude`”. Considerando que se usa GSM para la localización, las coordenadas no son exactas, pero son aproximadas. Se usa Google Maps para insertar automáticamente las coordenadas para su posterior envío al usuario.

```
if alerta == True:
    ciclo = 10
    #print "Modo Alerta"
    gsmlatitude, gsmlongitude = getCoordinates()
    response = 'Coordenadas de la maquina: https://maps.google.com/maps?q='
    + str(gsmlatitude) + ', ' + str(gsmlongitude)
    recv = cloud.sendSMS("+34 611 11 11 11", response)
    print recv
```

Figura 4.16: Sentencia condicional para enviar la localización

4.3.5. Envío de alertas al usuario

El usuario recibirá las alertas de dos maneras diferentes: vía *SMS* y correo electrónico. Para ello, se utilizará el módem *Hologram Nova* para tener conexión a Internet y enviar las alertas a la plataforma de *Hologram Cloud* (Servidor).

Para ello, en el código se escribe el mensaje que se quiere enviar a la nube y se envía junto con una etiqueta o tema (*topic*). Esto ayuda a distinguir en la nube si la alerta es de tipo *PIR*, giroscopio o localización ya que usa el mecanismo de rutas para rebotar el mensaje de la nube al usuario mediante *SMS* o correo electrónico. Por ejemplo, en la siguiente figura se puede ver cómo se configuran las alertas del *PIR*, del giroscopio y de la localización.

```
timex = strftime("%d-%m-%Y %H:%M:%S", gmtime()) #Creamos una cadena de texto con la hora
timex = timex + " MOVIMIENTO DETECTADO" #La sacamos por pantalla
print timex

recv = cloud.sendMessage(timex, topics = ['PIR_SENSOR'], timeout = 3)
print 'RESPONSE MESSAGE: ' + cloud.getResultString(recv)

timex = strftime("%d-%m-%Y %H:%M:%S", gmtime()) #Creamos una cadena de texto con la hora
timex = timex + " Sospecha de Robo" #La sacamos por pantalla
print timex
recv = cloud.sendMessage(timex, topics = ['GYRO_SENSOR'], timeout = 3)
print 'RESPONSE MESSAGE: ' + cloud.getResultString(recv)

recv = cloud.sendMessage(response, topics = ['Location'], timeout = 3)
print 'RESPONSE MESSAGE: ' + cloud.getResultString(recv)
```

Figura 4.17: Envío de mensaje con la etiqueta PIR_SENSOR, GYRO_SENSOR y Location

Cuando el mensaje llega a la nube, necesita ser encaminado a otro lugar ya que es necesario que el usuario sea consciente de la alerta. En la siguiente figura se puede observar que si llega un mensaje del módem conectado a la Raspberry Pi con el tema o etiqueta “PIR_SENSOR” o “GYRO_SENSOR”, entonces realiza la instrucción correspondiente:

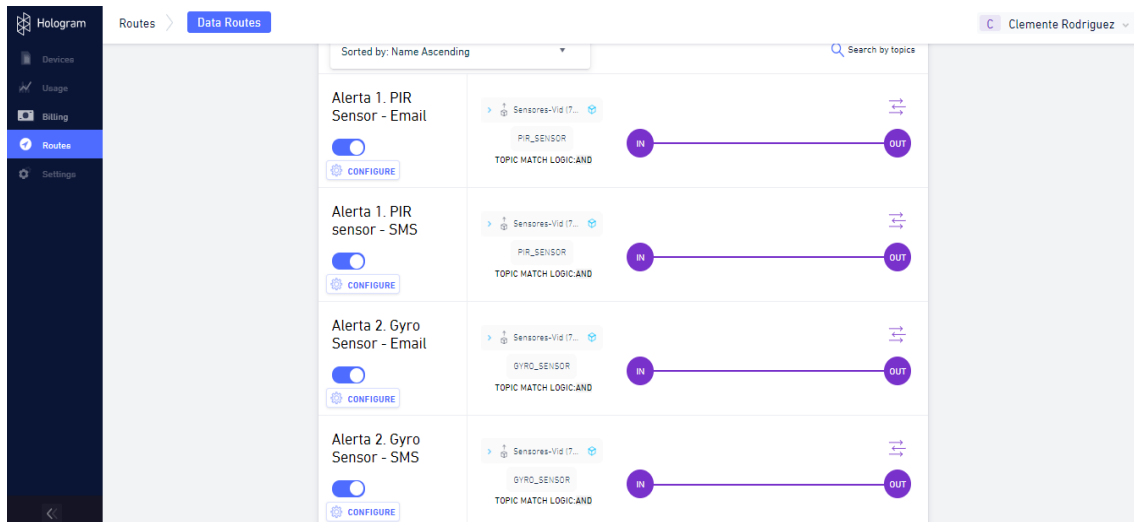


Figura 4.18: Página de rutas en Hologram Cloud

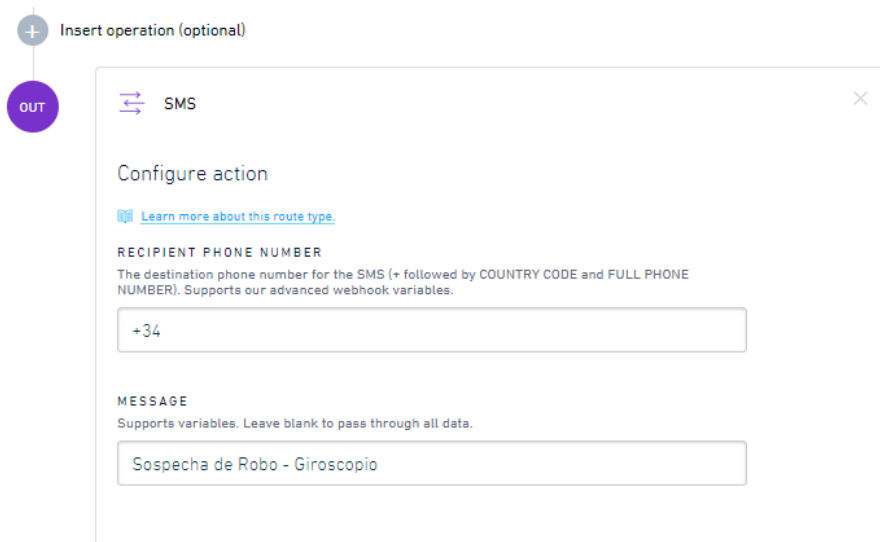


Figura 4.19: Ejemplo de configuración de ruta SMS

The image shows a configuration window for an 'Email' action in IFTTT. On the left, there is a vertical flow diagram with a purple circle labeled 'OUT' and a grey circle labeled 'Insert operation (optional)'. The main configuration area on the right is titled 'Email' and includes a 'Configure action' section with a link to 'Learn more about this route type'. Below this, there are three input fields: 'EMAIL RECIPIENTS' (with a placeholder 'Comma-separated list of email addresses'), 'SUBJECT' (with a placeholder 'Email subject'), and 'EMAIL MESSAGE' (with a placeholder '<<data>>' and a note 'Accepts our advanced webhook variables. (See docs link above)'). The 'SUBJECT' field contains the text 'Sospecha de Robo - Giroscopio'.

Figura 4.20: Ejemplo de configuración de ruta de correo electrónico

En este proyecto, existen 6 rutas en total: dos para las alertas del *PIR* y dos para las alertas del giroscopio (*SMS* y correo electrónico) y dos para la localización (*SMS*). Las rutas se configuran de la misma manera, aunque las etiquetas son diferentes.

4.3.6. Inicio de parámetros y librerías

En el programa de seguridad, se llaman a diferentes librerías y programas auxiliares. Como se ha comentado anteriormente, los programas auxiliares son `dueno.py` y `gyro.py`. En cuanto a las librerías, se usan las siguientes:

- ❖ **Time:** sirve para parar el programa con la función `sleep()` y conseguir la fecha y hora del momento y en la zona horaria correspondiente con `localtime()` y `strftime()`.
- ❖ **Hologram.HologramCloud:** útil en el uso del módem para enviar y recibir las notificaciones a la nube.
- ❖ **RPi.GPIO:** para el uso de los pines de la Raspberry Pi.
- ❖ **Smbus:** para vincular el giroscopio en el programa.
- ❖ **Math:** para realizar cálculos y operaciones trigonométricas con los datos obtenidos en el giroscopio.

4.3.7. Creación de archivos “log_seguridad.txt” y “alertas.txt”

En el caso de que ocurriese algún problema con la transmisión de los datos o se deseara ver qué movimientos ocurrieron en el pasado, existen dos archivos que contienen la información en la Raspberry Pi:

- ❖ **Log_seguridad.txt:** contiene toda la información que aporta el programa de seguridad. Todos los datos están acompañados de la fecha en la que se captaron. En el caso del giroscopio, mantiene el formato <fecha> <rotación> <aceleración>, mientras que el *PIR* tiene <fecha> <Movimiento detectado>.
- ❖ **Alertas.txt:** contiene solo las alertas ocasionadas por el *PIR* y el giroscopio. En el caso del *PIR*, el formato es el mismo que en log.txt, mientras que el giroscopio tiene <fecha> <Sospecha de robo – giroscopio>.

```
31-08-2019 19:05:52 - Rotacion: 0.261 1.843 - Aceleracion: -0.018 0.013
31-08-2019 19:05:54 - Rotacion: 0.439 0.819 - Aceleracion: -0.02 0.011
31-08-2019 19:05:56 - Rotacion: 0.146 1.02 - Aceleracion: -0.012 0.003
31-08-2019 19:05:58 - Rotacion: 0.276 1.686 - Aceleracion: -0.019 0.001
31-08-2019 19:06:00 - Rotacion: 0.231 0.043 - Aceleracion: -0.02 0.004
31-08-2019 19:06:02 - Rotacion: 0.101 0.101 - Aceleracion: -0.018 0.006
31-08-2019 19:06:04 - Rotacion: 0.029 0.979 - Aceleracion: -0.019 0.006
31-08-2019 19:06:06 - Rotacion: 0.436 0.93 - Aceleracion: -0.015 0.005
31-08-2019 19:06:08 - Rotacion: 0.216 0.894 - Aceleracion: -0.016 0.011
31-08-2019 19:06:10 - Rotacion: 0.118 1.719 - Aceleracion: -0.011 0.004
31-08-2019 19:06:12 - Rotacion: 0.246 1.015 - Aceleracion: -0.029 0.009
31-08-2019 19:06:14 - Rotacion: 0.087 1.857 - Aceleracion: -0.017 0.004
31-08-2019 19:06:16 - Rotacion: 0.232 1.26 - Aceleracion: -0.02 0.0
31-08-2019 19:06:18 - Rotacion: 0.392 1.294 - Aceleracion: -0.005 -0.011
31-08-2019 19:06:20 - Rotacion: 0.535 1.214 - Aceleracion: -0.02 0.009
```

Figura 4.21: Ejemplo log_seguridad.txt

```
28-08-2019 19:20:42 MOVIMIENTO DETECTADO - Sensor PIR
31-08-2019 19:25:13 MOVIMIENTO DETECTADO - Sensor PIR
31-08-2019 19:34:58 MOVIMIENTO DETECTADO - Sensor PIR
31-08-2019 20:29:48 MOVIMIENTO DETECTADO - Sensor PIR
31-08-2019 20:31:50 Sospecha de Robo - Giroscopio
31-08-2019 20:36:33 Sospecha de Robo - Giroscopio
```

Figura 4.22: Ejemplo alertas.txt

4.4. Terminal

En este apartado, el administrador podrá visualizar todas las opciones de monitorización del sistema de sensores a través de la terminal. Para ello, se realizará una conexión *SSH* para obtener toda la información que aporta la Raspberry Pi ya que está conectada en todo momento a Internet.

Al ser una conexión *SSH*, se tiene acceso total a la Raspberry Pi, desde parar o reiniciar los programas de riego y seguridad, a gestionar los recursos de la Raspberry Pi. Esto a la vez tiene un riesgo, si un usuario que no es el administrador logra introducirse en la Raspberry Pi puede controlarlo todo, hasta eliminar archivos y perder información privilegiada.

Sin embargo, realizar la conexión *SSH* no es una tarea trivial ya que el sistema se encuentra conectado a un módem USB y no es posible obtener una IP pública para acceder directamente. No obstante, gracias al servicio de Hologram, *Spacebridge*, es posible realizar dicha conexión. Consiste en crear un túnel seguro para enviar datos entre el módem *USB* y el ordenador que esté autenticado.

Para ello, el administrador descargarse la aplicación de *Spacebridge Client* y activar el servicio de *Spacebridge* seleccionando la SIM de la Raspberry Pi en la nube de Hologram:

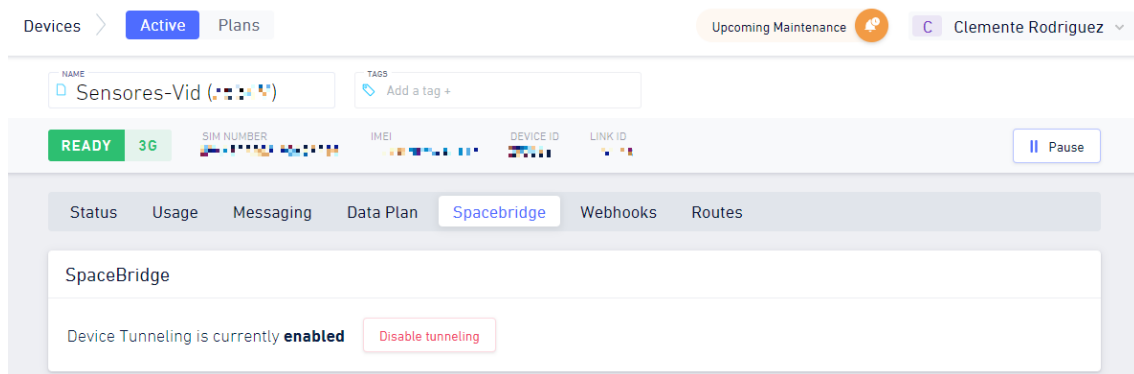


Figura 4.23: Activación de *Spacebridge* en la SIM seleccionada

En el ordenador, es necesario iniciar sesión con la clave de acceso de la cuenta de Hologram:

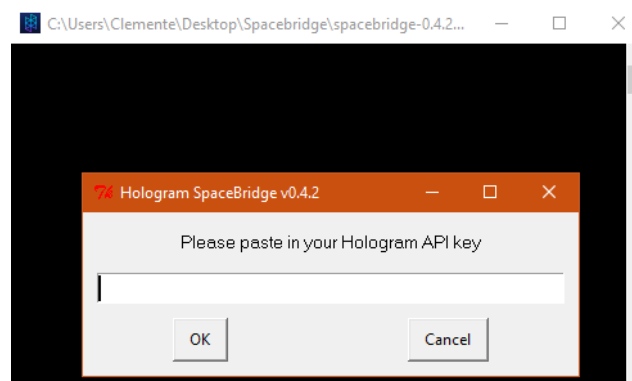


Figura 4.24: Inicio del programa *Spacebridge*

Una vez dentro, el usuario deberá seleccionar la SIM que tenga el servicio de *Spacebridge* activado e indicar a qué puertos estará conectado el túnel seguro. En este caso, al ser una futura conexión *SSH*, el puerto de la Raspberry Pi será el 22 y el puerto local puede ser uno cualquiera (más de 1023). En este caso, se ha elegido el puerto 12345:

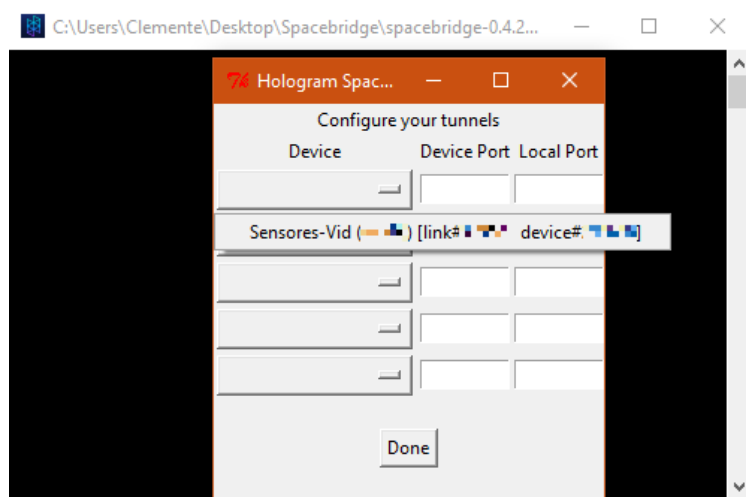


Figura 4.25: Selección de SIM y puertos

Si todo el proceso se ha realizado correctamente, debe aparecer una imagen como la siguiente figura. De esta manera, el túnel se ha creado:

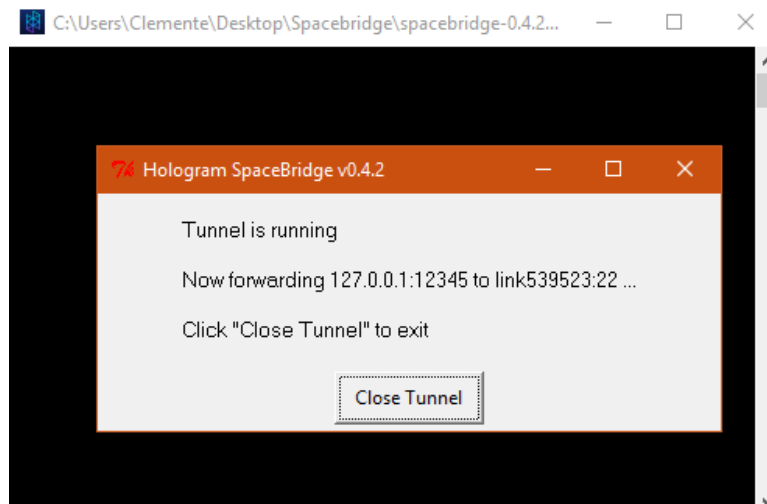


Figura 4.26: Túnel creado

Ahora, el usuario puede acceder por *SSH* a la Raspberry Pi. En este caso, se ha usado MobaXterm para acceder. Los parámetros son la IP del túnel, 127.0.0.1, y el puerto 12345. Una vez dentro, el administrador debe introducir usuario y contraseña de la

Raspberry Pi. Nótese que es necesario cambiar el usuario y contraseña por defecto que tiene la Raspberry Pi para iniciar sesión.

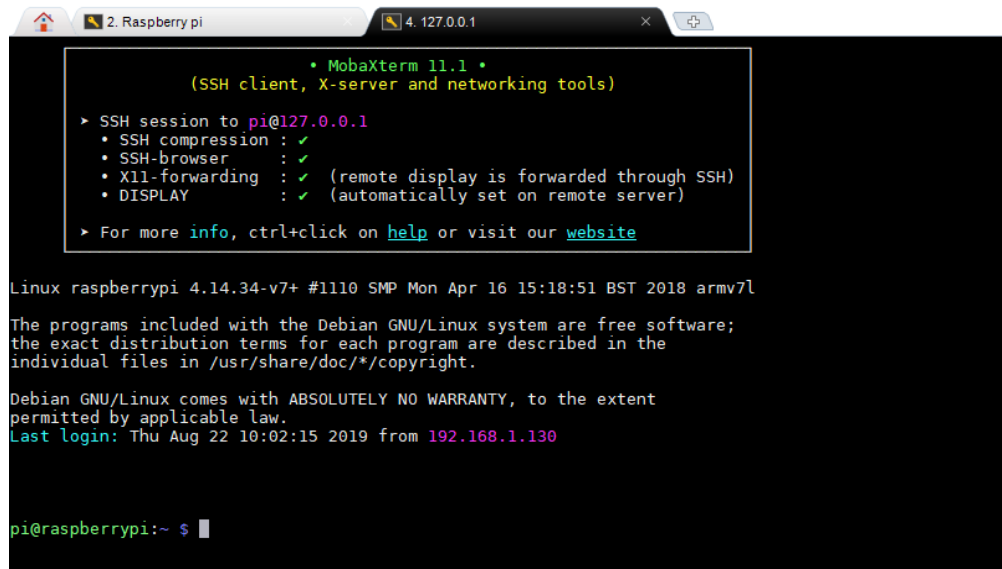


Figura 4.27: Acceso por *SSH*

Sin embargo, existe un problema con esta conexión. Es un poco más lenta ya que tiene conexión 3G, más lenta que si se hiciese con WiFi. Una ventaja que se consigue realizando el túnel y haciendo la conexión por *SSH* es que se consigue una doble seguridad. En primer lugar, para crear el túnel, primero se debe conocer la clave y que se encuentra en la página de Hologram (necesario iniciar sesión en la página). En segundo lugar, una vez creado el túnel, es necesario iniciar sesión en la Raspberry Pi y todas las comunicaciones están protegidas por *SSH*.

4.4.1. Comandos

Ya que se dispone de una conexión *SSH*, se puede controlar la Raspberry Pi como si estuviese en casa. El administrador puede desde iniciar o parar procesos hasta apagar o reiniciar la Raspberry Pi.

En este proyecto, el administrador tiene el deber de comprobar los archivos `log_seguridad.txt`, `log_riego.txt` y `alertas.txt` en el caso de que ocurriese algún problema o quisiese comprobar valores recogidos en el pasado. Por otro lado, también podrá acceder a la localización (aplicando el comando: `sudo hologram modem location`) y a los valores de la humedad. Además, deberá reiniciar o parar los

programas de seguridad y riego si se encuentran problemas con los datos. Finalmente, apagará la Raspberry Pi si es necesario un mantenimiento o un arreglo del sistema.

4.5. La unión de todas las piezas

Para iniciar todo el sistema en la Raspberry Pi, es necesario iniciar todos los programas al mismo tiempo, es decir, `riego.py` y `seguridad.py`. Además, hay que escribir el comando para iniciar la conexión a Internet por el módem.

En Linux (en este caso Raspbian), existe una herramienta para cumplir con este objetivo: Cron. Es un administrador de procesos en segundo plano que ejecuta procesos en diferentes intervalos de tiempo. Los procesos que deben ejecutarse están guardados en un fichero llamado “`crontab`”.

Sin embargo, antes de guardar los comandos en este fichero, es necesario otorgar permisos al sistema de lectura, escritura y ejecución de los programas creados anteriormente. Para ello, se escribe `chmod 777 <archivo.py>`, donde `archivo.py` son `riego.py` y `seguridad.py`.

Una vez hecho esto, se accede al fichero `crontab` con el comando “`crontab -e`” y se escribe lo siguiente:

```
@reboot sudo hologram network connect && sleep 10 && sudo
python2 /home/pi/Desktop/Hologram/seguridad.py && sleep 10
&& sudo python2 /home/pi/Desktop/Hologram/riego.py
```

En orden, esto significa que en cada arranque de la Raspberry Pi se activa la conexión del módem a Internet. Una vez hecho, se espera diez segundos y se inicia el programa de seguridad. Otros diez segundos después, se inicia el programa de riego. Para garantizar el funcionamiento, por seguridad se añaden 10 segundos entre las tareas.

4.6. Simulación

En este apartado, se muestra un ejemplo de uso del sistema en una viña con un motor de gasoil, cubriendo las principales acciones que se pueden realizar en el mismo. Para ello, se describirán paso a paso los diferentes eventos. A ser una simulación, solamente se quiere valorar el funcionamiento del sistema completo.

En primer lugar, la Raspberry Pi y la batería estarán colocadas en el lugar de trabajo, es decir, encima del motor de gasoil. El sensor de humedad tendrá un cable largo para que llegue al suelo y el sensor *PIR* se encontrará junto con el giroscopio en el motor. Este último estará mirando en el lado donde se encontrará el enganche del remolque, que es donde habría más presencia de personas en un caso sospechoso de robo.



Figura 4.28: Sistema de monitorización y el motor de gasoil



Figura 4.29: Sistema encima del motor de gasoil



Figura 4.30: Motor de gasoil con el sensor *PIR* en posición para detectar movimiento

Para iniciar el sistema, el cliente debe conectar la batería a la Raspberry Pi ya que cargarían los programas automáticamente gracias a “crontab”. Mientras se inician los programas, el cliente ya se habría ido para evitar que el sensor *PIR* detecte su presencia.

Para activar el sistema de seguridad, el cliente recibirá primero un SMS pidiendo que introduzca la clave que le haya llegado. Automáticamente, la seguridad se activará. Después de esto, el cliente solamente tendría que esperar hasta volver a echar gasoil al

motor. Al cabo de un tiempo, quiere ver cuál es el nivel de gasoil que hay en el depósito. Para ello, cuando llega al lugar de pruebas, envía el mensaje “D” y cuando reciba la respuesta de la Raspberry puede acceder al motor. Una vez haya terminado y quiera volver a casa, activa de nuevo el sistema enviando el mensaje “A”. Cuando reciba el mensaje de confirmación, el sistema de seguridad se ha vuelto a activar.

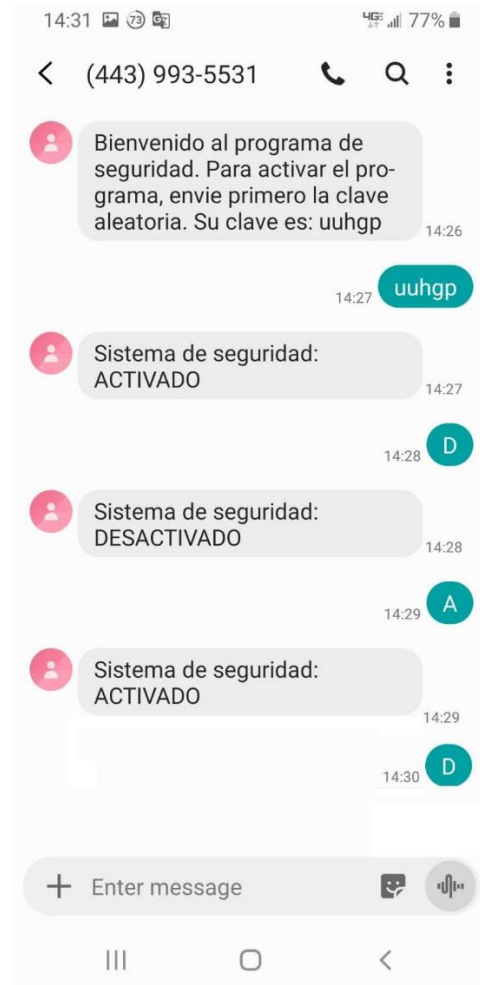


Figura 4.31: Bienvenida al programa y activación y desactivación del sistema de seguridad

Una vez en casa, si el cliente quiere ver cómo va la humedad del suelo, enciende el ordenador y se dirige a thingspeak.com. Después de iniciar sesión, con los permisos concedidos del administrador clicca en el canal donde se encuentra la información de la humedad.

Simulando que saltan las alarmas gracias al PIR y al giroscopio, el cliente recibe a la vez un SMS y un correo electrónico diciendo qué sensor ha saltado junto con la localización del dispositivo. Inmediatamente, si ve que la localización no es la del lugar de pruebas, avisaría a las autoridades e informaría de la situación hasta encontrar la máquina. Si todo se ha resuelto, el cliente debe desactivar y activar el sistema de seguridad utilizando los SMS.

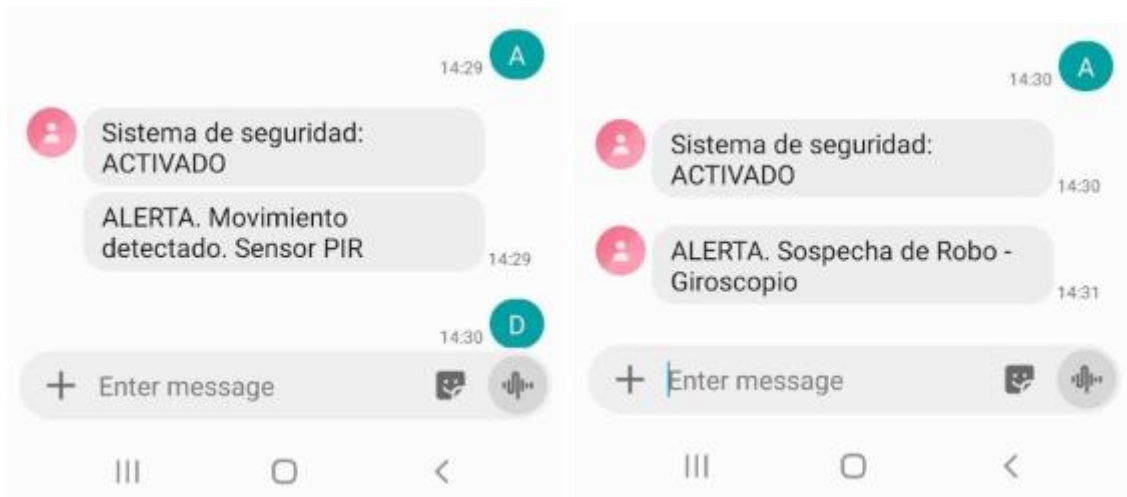


Figura 4.32: Alertas vía SMS



Figura 4.33: Alerta y envío de localización

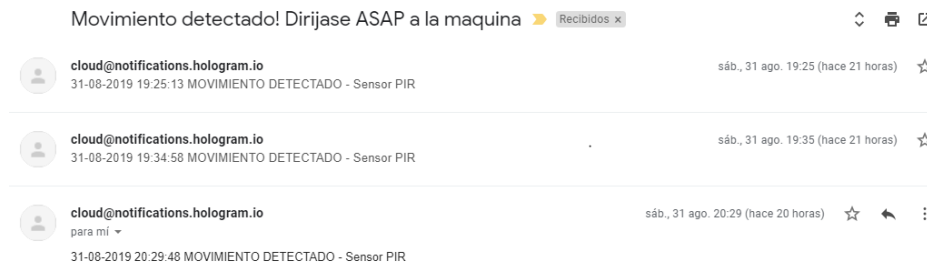


Figura 4.34: Notificaciones por correo electrónico – sensor *PIR*

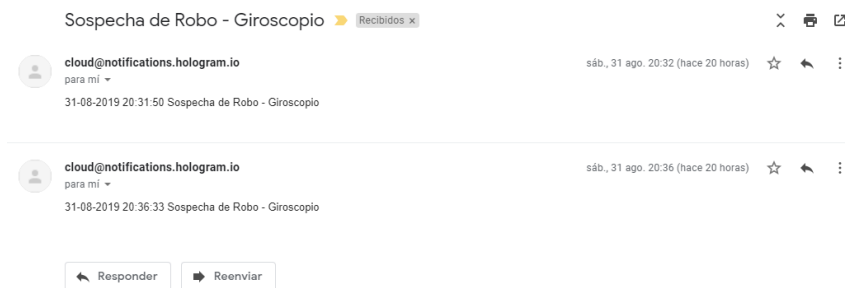


Figura 4.35: Notificaciones por correo electrónico – giroscopio

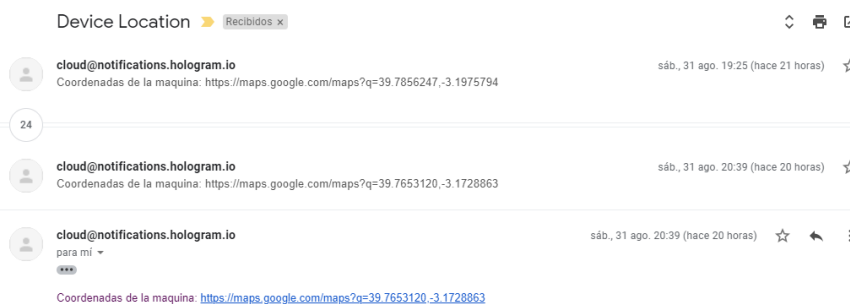


Figura 4.36: Notificaciones por correo electrónico – localización

Por otro lado, si hubiese un malfuncionamiento del sistema de seguridad o de riego, el cliente avisaría al administrador y este revisaría el sistema utilizando la terminal. En primer lugar, miraría los archivos `log_seguridad.txt`, `log_riego.txt` y `alertas.txt` para ver si hay una actividad anómala. Si es un problema de software, pararía, revisaría y volvería a activar los programas y si fuese un problema de hardware, apagaría la Raspberry Pi a distancia y examinaría el sistema in situ.

Capítulo 5

Pruebas

En este capítulo, se van a realizar las pruebas de evaluación, tanto en el laboratorio como en un escenario real, de los requisitos funcionales del sistema. A continuación, se va a determinar si los requisitos se han cumplido en la implementación del producto:

- ❖ **RSF-01. Valor de la humedad en el suelo en tiempo real:** como se puede ver en `riego.py`, se recogen valores de la humedad cada 15 minutos. Por lo tanto, se cumple con este requisito correctamente.
- ❖ **RSF-02. Valor de la aceleración en tiempo real:** en `seguridad.py`, se recogen valores de la aceleración en los ejes X e Y cada 2 segundos a menos que esté desactivado o en modo alerta. Por lo tanto, se cumple con este requisito correctamente.
- ❖ **RSF-03. Valor de la rotación en tiempo real:** en `seguridad.py`, se recogen valores de la rotación en los ejes X e Y cada 2 segundos a menos que esté desactivado o en modo alerta. Por lo tanto, se cumple con este requisito correctamente.
- ❖ **RSF-04. Detección de cuerpo térmico:** en `seguridad.py`, se comprueba si hay variación de las radiaciones infrarrojas cada 2 segundos a menos que esté desactivado o en modo alerta. Por lo tanto, se cumple con este requisito correctamente.

- ❖ **RSF-05. Registro de los datos a lo largo del tiempo:** en el caso de la humedad, existe un archivo “log_riego.txt” donde se guardaba la actividad del programa junto con los valores de humedad y, además, se almacena la información de la humedad en la interfaz web de *ThingSpeak* en forma de gráfico o de archivo CSV, JSON o XML. En cuanto a los valores de la seguridad, se encuentran guardados en los archivos “log_seguridad.txt” donde se guarda la actividad del programa junto con los valores de los sensores y “alertas.txt” donde se guardan solamente las alertas. Por lo tanto, se cumple con este requisito correctamente.
- ❖ **RSF-06. Interfaz web:** debido a las limitaciones en cuanto al uso de datos de Hologram (1 MB mensual) solamente se enviarían los datos de la humedad a ThingSpeak ya que es el dato más importante para ver en un gráfico. Se cumple parcialmente con el requisito.
- ❖ **RSF-07. Terminal:** gracias a una conexión SSH, se puede acceder totalmente al sistema y, por lo tanto, a los archivos log que tienen la información de los sensores. Adicionalmente, se puede parar o iniciar procesos y apagar o reiniciar el dispositivo a larga distancia. Por lo tanto, se cumple muy bien este requisito.
- ❖ **RSF-08. Comunicación del sistema:** los dispositivos se pueden comunicar con el usuario de tres maneras diferentes: con *ThingSpeak*, con SMS y correos electrónicos de alerta y utilizando la terminal. Sin embargo, debido a las limitaciones del servicio de Hologram, no es posible enviar SMS a teléfonos españoles. En base a las pruebas, se consiguió enviar SMS a teléfonos de Reino Unido y Estados Unidos. De todas maneras, gracias al módem Hologram Nova, es posible cumplir correctamente este requisito en el resto de los aspectos.
- ❖ **RSF-09. Activación o desactivación del sistema de seguridad:** con un SMS, el usuario tiene el poder de activar o desactivar el sistema de seguridad. Siguiendo la misma razón que en el requisito funcional 8, no es posible adquirir un número de teléfono español para la SIM en la Raspberry Pi y enviar SMS ahí. Como alternativa, se puede adquirir un número que funcione en otro país (como en Estados Unidos) y enviar SMS allí o enviarlos gratuitamente desde la nube de Hologram. Se cumple parcialmente con este requisito.
- ❖ **RSF-10. Activación o desactivación del motor de gasoil:** en riego.py, el motor se enciende o se apaga en función de las condiciones de humedad en la zona. Por lo tanto, el requisito se cumple correctamente.

- ❖ **RSF-11. Comunicación automática vía correo electrónico:** la comunicación por correo electrónico es efectiva cuando se produce una alerta. Por lo tanto, el requisito se cumple correctamente.
- ❖ **RSF-12. Comunicación automática vía SMS:** como ya se ha explicado en otros requisitos, la comunicación vía SMS no funciona correctamente en España debido a las limitaciones del servicio de la tarjeta SIM que se está utilizando. Sin embargo, en las pruebas se ha conseguido comunicación utilizando un teléfono estadounidense. Por lo tanto, el requisito se cumpliría parcialmente.

Esta información se puede resumir en la Tabla 5.1:

Requisito	Resultado	Observaciones
RSF-01	Correcto	***
RSF-02	Correcto	***
RSF-03	Correcto	***
RSF-04	Correcto	***
RSF-05	Correcto	***
RSF-06	Funciona parcialmente	Limitaciones de datos móviles (1 MB)
RSF-07	Correcto	***
RSF-08	Funciona parcialmente	Limitaciones servicio en España
RSF-09	Funciona parcialmente	Limitaciones servicio en España
RSF-10	Correcto	***
RSF-11	Correcto	***
RSF-12	Funciona parcialmente	Limitaciones servicio en España

Tabla 5.1: Resumen de las pruebas

Capítulo 6

Planificación y presupuesto

En este capítulo, se expone la planificación del trabajo mostrando las diferentes fases y sus fechas de inicio y finalización. Además, se listarán los recursos utilizados, sus costes asociados y el coste total del proyecto.

6.1. Planificación

El desarrollo de este proyecto ha seguido diferentes fases: planificación previa, la definición del problema, el estudio de conceptos necesarios, el diseño, la implementación y la veracidad de su funcionamiento. Dada la longitud del proyecto, era necesario tener en cuenta el tiempo para su realización.

Una ventaja que se tenía en este proyecto es que se había elegido el problema con mucho tiempo. Esto permitía que se reflexionase sobre todos los puntos posibles a realizar ya que a pesar de que el problema principal estuviese muy claro, eran muchas las ideas para abordarlo. De hecho, a lo largo de su desarrollo, se cambiaron algunas herramientas para resolver el problema. A continuación, se incluirá una tabla que mostrará las horas totales de dedicación en este proyecto. En la misma, una columna extra indicará el número de horas estimadas diarias ya que no se invertía el mismo tiempo en diferentes épocas del

año debido a otros trabajos o evaluaciones de carácter universitario. Además, después de la tabla, se incluirá un diagrama de Gantt para visualizar la información de la tabla:

Actividad	Inicio estimado (Día/Mes/Año)	Fin estimado (Día/Mes/Año)	Número días estimado	Estimación horas/día	Horas totales actividad
Estudio Raspberry Pi – Instalaciones – Raspbian OS	07/08/2018	18/08/2018	11	1	11
Planificación – Estudio del entorno	25/09/2018	02/10/2018	7	2	14
Definición del problema	01/10/2018	05/10/2018	4	1	4
Estudio sensor humedad	17/03/2019	19/03/2019	2	2	4
Estudio giroscopio	02/04/2019	04/04/2019	2	2	4
Estudio <i>PIR</i>	16/04/2019	18/04/2019	2	2	4
Planificación – Investigación <i>IoT</i>	15/05/2019	23/05/2019	8	2,5	20
Definición del problema – Modificación de objetivos	01/06/2019	07/06/2019	6	2,5	15
Creación de programas – riego.py y seguridad.py	20/06/2019	29/06/2019	9	2,5	22,5
Estudio de Hologram Nova	28/06/2019	10/07/2019	12	2,5	30
Diseño del sistema completo	20/07/2019	25/07/2019	5	2	10
Modificación de los programas – Unión con Hologram	28/07/2019	05/08/2019	8	2	16
Simulaciones	01/08/2019	12/08/2019	11	1	11
Verificación y correcciones	15/08/2019	24/08/2019	9	1	9
Propuestas de mejora y conclusiones	28/08/2019	30/08/2019	2	1	2
Documentación	01/06/2019	31/08/2019	91	2,5	227,5
		Días totales	127	Horas totales	404

Tabla 6.1: Evolución del proyecto

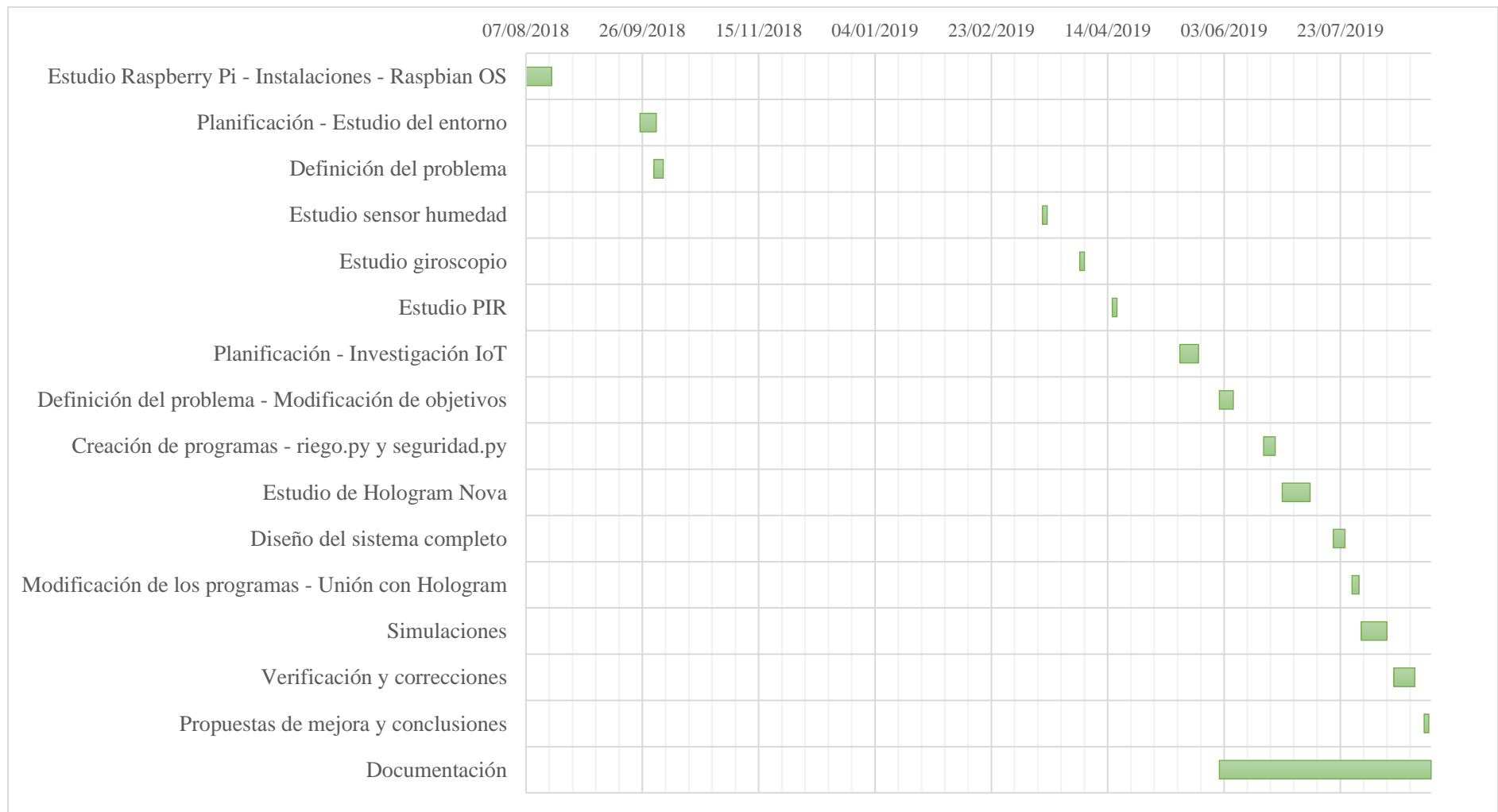


Figura 6.1: Diagrama de Gantt

6.2. Presupuesto

En este apartado, se expondrá el coste total del proyecto. Este consiste en costes relativos de personal contratado, el coste de los materiales utilizados y otros costes.

6.2.1. Coste de materiales

En cuanto a los costes materiales, se tendrá en cuenta una tasa de amortización que determina el coste final de un material en función de su tiempo utilizado en el proyecto:

$$\text{Tasa de Amortización} = \frac{A}{B} \cdot C \cdot D$$

Donde:

- ❖ A: Número de meses desde la fecha de facturación en el que el equipo es utilizado.
- ❖ B: Periodo de depreciación (60 meses)
- ❖ C: Coste del equipo sin IVA (21%)
- ❖ D: Porcentaje del uso que se dedica al proyecto (Generalmente es el 100%)

Producto	Precio (€)	Precio sin IVA (€)	Tiempo dedicado (meses)	Coste final (€)
Ordenador HP Pavilion	900,00	711,00	11	130,35
Teléfono One Plus 6T	480,00	379,20	8	50,56
Raspberry Pi 3 Model B *	36,30	28,68	11	5,25
RPi GPIO Breakout Board	7,99	6,31	11	1,16
Sensor PIR HC-SR501 *	1,80	1,42	8	0,19
Giroscopio MPU6050 *	2,70	2,13	8	0,28
Sensor humedad del suelo *	2,40	1,89	8	0,25
Convertidor analógico a	5,05	3,99	2	0,14

digital MCP3008				
Tarjeta SIM - Hologram	\$ 8,77 (8,05 €)	8,05	4	0,54
Módem Hologram Nova	\$ 69,00 (65,00 €)	65,00 €	4	4,33
Protoboard	3,00	2,37	11	0,43
Cables hembra - hembra	1,61	1,27	5	0,11
Pack Luces Led	1,87	1,48	2	0,05
Pack Resistencias 1K Ω	0,50	0,40	8	0,05
	Coste (€)	1.213,19	Coste total (€)	193,69

Tabla 6.2: Coste de los materiales

Nota: No se aplica el Impuesto del Valor Añadido para los productos que se compraron en dólares. En estos casos, se pagaron impuestos por aduanas que se contabilizarán en otros gastos. Por otro lado, los productos marcados con un asterisco (*) fueron prestados por el departamento de Telemática de la Universidad Carlos III Madrid.

6.2.2. Otros costes

A continuación, se muestra una lista otros costes derivados del proyecto:

Producto	Precio (€)	Precio sin IVA (€)
Transporte - Módem Hologram Nova	38,31	30,26
Aduanas – IVA aplicado a Módem Hologram Nova	19,24	0,00
Aduanas - Gestión de despacho	19,36	16,00
Gestión número de teléfono Hologram	\$ 1,00 (0,92 €)	0,92 €
	Coste total (€)	47,18 €

Tabla 6.3: Otros costes

6.2.3. Coste de personal

Se ha realizado un estudio del presupuesto como si de un proyecto real en un entorno corporativo se tratase, dividiendo el proyecto dos puestos: el jefe de proyecto, que en este caso se reflejaría como el tutor, y el desarrollador, como el estudiante.

Puesto	Salario estimado (anual bruto, €)	Salario por hora (1755 horas anuales [95])	Horas totales	Salario Bruto total	Seguridad Social (6.2 %)	Otras deducciones (estimación 4 %)	Salario neto total
Jefe de Proyecto	33.762,00	19,23	76	1.462,05	90,64	58,48	1312,92
Desarrollador	25.304,00	14,41	328	4.729,18	293,20	189,17	4.246,80
			Total bruto (€)	6.191,23		Total neto (€)	5.559,73

Tabla 6.4: Salarios de los empleados

6.2.4. Coste total del proyecto

Para calcular el coste total del proyecto es necesario sumar todos los costes. Para ello, primero se procederá a sumar los costes materiales y los de otros costes y aplicarles el IVA.

Descripción	Coste (€)	Coste final (21 % IVA, €)
Materiales	193,69	234,36
Otros	47,18	57,08 + 19,24
	Coste total (€)	310,68

Tabla 6.5: Coste aplicando el IVA

Nótese que se añaden los 19,24 € del IVA de aduanas de Hologram Nova. Por lo tanto, el coste final del proyecto será:

$$\text{Coste final} = \text{Coste}_{\text{Materiales}} + \text{Coste}_{\text{Personal}} = 310,68 + 6.191,23 = 6.501,91 \text{ €}$$

El presupuesto final del proyecto es de **SEIS MIL QUINIENTOS UNO CON NOVENTA Y UNO EUROS (6.501,91 €)**

Capítulo 7

Marco Regulador

A la hora de desarrollar nuevas tecnologías para favorecer la economía digital y el acceso más rápido a la información, se deben seguir unas reglas para que la privacidad del usuario se respete en todo momento.

7.1. Normas y legislación

En este apartado, se expondrán las normas que se han seguido en este proyecto con el objetivo de mejorar la experiencia del usuario y proteger su privacidad:

- ❖ **Ley Orgánica 3/2018 de Protección de Datos Personales y garantía de los derechos digitales** [96]: Tuvo lugar con la adopción de **la Ley Europea de Protección de Datos** [97] y esta ley busca proteger la seguridad los datos personales de los clientes. En el proyecto, la información de los sensores y la localización del dispositivo está protegida ya que, para acceder a ella, siempre es necesario iniciar sesión (sea por la plataforma de ThingSpeak o la terminal). Además, la información del cliente en relación con el número de teléfono y el correo electrónico se almacena de manera segura en el servidor de Hologram utilizando encriptación SSL. En este último punto, Hologram obedece las leyes

de protección europeas ya que no existe una legislación definida en el mismo ámbito en Estados Unidos.

- ❖ **Decisión de ejecución (UE) 2016/1250 de la Comisión con arreglo a la Directiva 95/46/CE del Parlamento Europeo y del Consejo, sobre la adecuación de la protección conferida por el Escudo de la privacidad UE-EE.UU.** [98]. Aquí se establecen las normas en el traspaso de datos entre la Unión Europea y Estados Unidos. Esta ley es importante en el proyecto ya que se está utilizando Hologram [99], un servicio estadounidense en el cual se depositan los datos personales del cliente para que reciba las alertas.
- ❖ **Real Decreto 1078/2014** [100]: Se establecen las normas que deben cumplir los agricultores para poder recibir las ayudas económicas de la Política Agraria Común o PAC. Uno de los requisitos es que se mantenga la tierra en buen estado. En el proyecto, se asume que la tierra debe estar en buen estado para hacer las mediciones correctamente.

7.2. Licencias y tecnologías

A continuación, se citan las licencias de uso referentes a los programas y tecnologías empleados en el desarrollo del proyecto:

- ❖ **Microsoft Windows 10:** La licencia es de tipo Home, cedida gratuitamente por Microsoft en la transición de Windows 8 a 10.
- ❖ **Microsoft Office 365:** El paquete de ofimática empleado para el desarrollo del proyecto cuya licencia es de tipo estudiante, está cedida por Microsoft gracias a la colaboración de la Universidad Carlos III de Madrid.
- ❖ **MobaXTerm:** terminal compatible en Windows con herramientas de red para computación remota con licencia gratuita.
- ❖ **Raspbian OS:** El sistema operativo integrado en la Raspberry Pi con licencia *GNU*.
- ❖ **Python:** El lenguaje es administrado por Python Software Foundation, que tiene la licencia de código abierto llamado Python Software Foundation License.
- ❖ **RStudio:** El *IDE* para el lenguaje de programación R mantiene una licencia libre para uso no comercial.

- ❖ **Hologram Cloud:** La inspección del flujo de datos por el módem de la marca Hologram es gratuita ya que se asumen los costes de las tarifas de la tarjeta SIM y del módem.
- ❖ **Draw.io:** La aplicación web de creación de diagramas es de código abierto.

7.3. Propiedad Intelectual

El proyecto ha sido una exploración de diversas tecnologías para la creación de un sistema completo. Así pues, su distribución es libre siempre y cuando se referencie al presente trabajo y al autor de este.

Los programas y los ficheros relacionados con el proyecto se encuentran en la cuenta de GitHub. Se pueden descargar desde la siguiente dirección:
<https://github.com/clementearraez/sensors>.

Capítulo 8

Conclusiones y líneas futuras

En este capítulo se mostrarán las conclusiones del proyecto y aquellas líneas futuras que podrían mejorarlo.

8.1. Conclusiones

Este proyecto se enfoca en aplicar los conceptos de internet de las cosas y de redes de sensores para mejorar el acceso de los usuarios a la información de un sistema de monitorización. El trabajo se adapta muy bien a la época actual ya que existe una tendencia a automatizar y digitalizar las cosas del entorno, es decir, a acceder a la información más rápido y sencillo.

Como ya se ha expuesto en el resto del documento, para optimizar el riego automático y mejorar la seguridad de las máquinas agrícolas, se han utilizado tres escenarios: El acceso a la información de la humedad en tiempo real gracias a la plataforma web, el envío de alertas de robo al usuario vía SMS y correo electrónico gracias al servidor de Hologram y el uso de la terminal para tener a toda la información de los procesos en el lugar de pruebas.

El desarrollo del proyecto ha sido bastante interesante ya que se han utilizado diferentes dispositivos y tecnologías que nunca se había visto antes. Esto implica que se necesite más tiempo para pensar cómo utilizar bien todas las herramientas ya que hubo casos en los que se tuvo que cambiar de idea. Por ejemplo, antes se pensaba utilizar una conexión TCP en la terminal y se pasó haciendo una conexión SSH, mejorando la experiencia del usuario. Continuando con esto, uno de los retos que se tenía desde el principio era la comunicación a larga distancia entre el sistema de sensores y el ordenador de casa. Gracias al uso del módem de Hologram y de la numerosa cantidad de servicios disponibles en la nube, el traspaso de información se hizo de manera muy interactiva.

El nivel de exigencia y dedicación que requería este proyecto era alto, pero ha sido muy satisfactorio. Lo bueno de este trabajo es que la idea principal estaba muy clara desde el principio ya que la idea de ayudar a aquellos agricultores que invierten mucho tiempo para sacar su economía adelante es muy motivadora, aunque al principio no se tenían los conocimientos suficientes para conseguirlo. Gracias a la labor de la tutora y de resolver muchos problemas junto con buenos amigos, este proyecto llegó a ser posible. Además, es motivo de orgullo haber propuesto una nueva forma de ver el trabajo agrario.

8.2. Líneas futuras

Entre las muchas mejoras que se podrían realizar en el futuro, se podrían destacar las siguientes:

- **Uso de información meteorológica en tiempo real**, en lugar de utilizar estadísticas pasadas para el cálculo del agua disponible diaria (anexo 2). Se podrían usar datos de meteorología como la agencia estatal de meteorología (AEMET) para saber en el momento cuál sería la necesidad de agua diaria ya que el nivel de la lluvia no es el mismo todos los años. De esa manera, analizando la necesidad mensual y las probabilidades de lluvia diarias, se optimizaría aún más el riego automático.
- **Despliegue de una red de sensores para ampliar la extensión de las mediciones**. Se podría conseguir esto aumentando el número de nodos sensores de humedad (ya que el de seguridad estaría en el motor) y utilizar sistemas de comunicación como ZigBee en función del tamaño del terreno a analizar.

Además, se podría cubrir otras tierras que tengan otros motores y controlar todas al mismo tiempo con el mismo sistema.

- Como tercera mejora, mucha gente está empezando a utilizar placas solares para regar las plantas, aunque por la noche no se tenga la energía del sol para regar. También se podría **enfocar el sistema en el uso de la energía solar** para favorecer un desarrollo más ecológico.
- **Alimentación de la batería y mantenimiento del sistema.** En cuanto a la batería, sería conveniente encontrar la manera de alimentarla gracias a un motor de gasoil o la placa solar para que el sistema pueda funcionar a todas horas. Al mismo tiempo, convendría colocar correctamente el hardware desplegado en algún recipiente resistente para protegerlo de los agentes externos como las lluvias o bajas o altas temperaturas, animales o de intentos de robo.

Referencias

- [1] "Historia y Evolución de la Tecnología", *Areatecnologia.com*, 2019. [En línea]. Disponible: <https://www.areatecnologia.com/historia-evolucion-tecnologia.htm>. [Acceso: 25- Sep- 2018]
- [2] "IoT Connections to Grow 140% to Hit 50 Billion By 2022", *Juniperresearch.com*, 2019. [En línea]. Disponible: <https://www.juniperresearch.com/press/press-releases/iot-connections-to-grow-140-to-hit-50-billion>. [Acceso: 25- May- 2019]
- [3] D. CCH, "Tutorial Estrategias de Aprendizaje", *Estrategias de Aprendizaje*, 2019. [En línea]. Disponible: <http://tutorial.cch.unam.mx/bloque4/lasTIC>. [Acceso: 26 - Sep - 2018]
- [4] M. Mela, "¿Qué son las TIC y para qué sirven? | Noticias Iberestudios", *Noticias.iberestudios.com*, 2019. [En línea]. Disponible: <http://noticias.iberestudios.com/%C2%BFque-son-las-tic-y-para-que-sirven/>. [Acceso: 14 - May - 2019]
- [5] S. Adamala, N. Raghuwanshi and A. Mishra, "Development of Surface Irrigation Systems Design and Evaluation Software (SIDES)", *Computers and Electronics in Agriculture*, vol. 100, pp. 100-109, 2014. [Acceso: 17 - May - 2019]
- [6] "GREENPOWER DEUTZ Diesel Power generator 30kVA 24kW Open frame Manual starting", *Greenpower.lk*, 2019. [En línea]. Disponible: <https://www.greenpower.lk/katalog/artikel/greenpower-deutz-diesel-power-generator-30kva-24kw-open-frame-manual-starting/116586-0-50/>. [Acceso: 17 - May - 2019]
- [7] "OLX Argentina", *Help.olx.com.ar*, 2019. [En línea]. Disponible: <https://help.olx.com.ar/hc/es-419>. [Acceso: 6 - Jun - 2019]

- [8] N. López, "Desmantelado un grupo criminal en Castilla-La Mancha especializado en el robo de motores de riego - El Digital de Albacete", *El Digital de Albacete*, 2019. [En línea]. Disponible: <https://www.eldigitaldealbacete.com/2017/05/18/desmantelado-grupo-criminal-castilla-la-mancha-especializado-robo-motores-riego/>. [Acceso: 6 - Jun - 2019]
- [9] "ElectroCrea: Mayorista en electrónica", *ElectroCrea*, 2019. [En línea]. Disponible: <https://electrocrea.com/>. [Acceso: 6 - Jun - 2019]
- [10] "Medir la humedad del suelo con Arduino e higrómetro FC-28", *Luis Llamas*, 2019. [En línea]. Disponible: <https://www.luisllamas.es/arduino-humedad-suelo-fc-28/>. [Acceso: 25 - Jun - 2019]
- [11] "MCP3008 - 8-Channel 10-Bit ADC With SPI Interface", *The Pi Hut*, 2019. [En línea]. Disponible: <https://thepihut.com/products/adafruit-mcp3008-8-channel-10-bit-adc-with-spi-interface>. [Acceso: 25 - Jun - 2019]
- [12] "Interfacing Soil Moisture Sensor with Raspberry Pi | Raspberry Pi", *Maker Pro*, 2019. [En línea]. Disponible: <https://maker.pro/raspberry-pi/tutorial/interfacing-soil-moisture-sensor-with-raspberry-pi>. [Acceso: 25 - Jun - 2019]
- [13] "PIR Motion Detection Sensor HC-SR501 [SEN-2511] : rhydoLABZ INDIA", *rhydoLABZ.com*, 2019. [En línea]. Disponible: https://www.rhydolabz.com/sensors-ir-pir-sensors-c-137_150/pir-motion-detection-sensor-hcsr501-p-1512.html?zenid=3ccvrqqaigdusb8qf17bgn6v93. [Acceso: 28 - Jun - 2019]
- [14] *Cdn-learn.adafruit.com*, 2019. [En línea]. Disponible: <https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf>. [Acceso: 28 - Jun - 2019]
- [15] "Accelerometer and Gyro Breakout MPU6050 Philippines", *Makerlab-electronics.com*, 2019. [En línea]. Disponible: <https://www.makerlab-electronics.com/product/triple-axis-accelerometer-gyro-breakout-mpu6050/>. [Acceso: 29 - Jun - 2019]
- [16] "Hologram Nova User Manual", *Hologram*, 2019. [En línea]. Disponible: <https://hologram.io/docs/reference/nova/user-manual/>. [Acceso: 01 - Jul - 2019]

- [17] N. Gershenfeld, R. Krikorian and D. Cohen, "The Internet of Things. The principles that gave rise to the Internet are now leading to a new kind of network of everyday devices, an "Internet-0"', *Scientific American*, 2019 [En línea]. Disponible: <http://cba.mit.edu/docs/papers/04.10.i0.pdf>. [Acceso: 4 - Jun - 2019]
- [18] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions", *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645-1660, 2013. [Acceso: 4 - Jun - 2019]
- [19] "Internet of Things | Europe.Hughes.com", *Europe.hughes.com*, 2019. [En línea]. Disponible: <https://europe.hughes.com/sectors/utilities/internet-things>. [Acceso: 4 - Jun - 2019]
- [20] "Internet De Las Cosas para mejorar la productividad | TicNegocios | Cámara Valencia", *Tecnología para los negocios*, 2019. [En línea]. Disponible: <https://ticnegocios.camaravalencia.com/servicios/tendencias/internet-las-cosas-mejorar-la-productividad-empresa/>. [Acceso: 4 - Jun - 2019]
- [21] "How the Internet of Things will Impact our Productivity", *InfoQ*, 2019. [En línea]. Disponible: <https://www.infoq.com/articles/iot-impact-productivity/>. [Acceso: 24 - Jun - 2019]
- [22] "Auto-ID Labs", *En.wikipedia.org*, 2019. [En línea]. Disponible: https://en.wikipedia.org/wiki/Auto-ID_Labs. [Acceso: 24 - Jun - 2019]
- [23] "La historia detrás de la internet de las cosas | Elespectador.com", *Elespectador.com*, 2019. [En línea]. Disponible: <https://www.elespectador.com/tecnologia/la-historia-detras-de-la-internet-de-las-cosas-articulo-716678>. [Acceso: 24 - Jun - 2019]
- [24] "Origen e historia del Internet of Things – Máster en Internet of Things", *Master-internet-of-things.com*, 2019. [En línea]. Disponible: <https://www.master-internet-of-things.com/historia-iot/>. [Acceso: 24 - Jun - 2019]
- [25] E. Iglesia, "La relación de IoT con Big Data", *Campusbigdata.com*, 2019. [En línea]. Disponible: <https://www.campusbigdata.com/big-data-blog/item/101-relacion-iot-con-big-data>. [Acceso: 02- Sep- 2019]
- [26] M. Hung, *Leading the IoT. Gartner Insights on How to Lead in a Connected World*. 2017 [En línea]. Disponible:

https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf. [Acceso: 25 - Jun - 2019]

[27] J. Waldner, *Nano-Informatique & Intelligence Ambiante - Inventer l'Ordinateur du XXIeme Siècle*. Londres: Hermes Science, 2007, p. 254.

[28] "State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating", *Iot-analytics.com*, 2019. [En línea]. Disponible: <https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/>. [Acceso: 25 - Jun - 2019]

[29] "Número de dispositivos conectados usados en España 2017 | Statista", *Statista*, 2019. [En línea]. Disponible: <https://es.statista.com/estadisticas/481066/numero-de-dispositivos-conectados-por-persona-en-espana/>. [Acceso: 6 - Jun - 2019]

[30] D. de la Torre, "Las 10 tendencias y datos más importantes de IoT", *Blogthinkbig.com*, 2019. [En línea]. Disponible: <https://blogthinkbig.com/tendencias-iot-futuro>. [Acceso: 24 - Jun - 2019]

[31] "Forrester : Internet of Things (IoT)", *Forrester.com*, 2019. [En línea]. Disponible: <https://www.forrester.com/Internet-Of-Things-%28IoT%29>. [Acceso: 27 - Jun - 2019]

[32] A. Manjarrés, "El futuro del IoT y los riesgos que conlleva", *Blogs.deusto.es*, 2019. [En línea]. Disponible: <https://blogs.deusto.es/master-informatica/el-futuro-del-iot-y-los-riesgos-que-conlleva/>. [Acceso: 27 - Jun - 2019]

[33] M. Torchia and M. Shirer, "IDC Forecasts Worldwide Spending on the Internet of Things to Reach \$745 Billion in 2019, Led by the Manufacturing, Consumer, Transportation, and Utilities Sectors", *IDC: The premier global market intelligence company*, 2019. [En línea]. Disponible: <https://www.idc.com/getdoc.jsp?containerId=prUS44596319>. [Acceso: 2 - Jun - 2019]

[34] "Enabling the Smart Agriculture Revolution - The Future of Farming through the IoT Perspective :: Beecham Research", *Beechamresearch.com*, 2016. [En línea]. Disponible: <http://www.beechamresearch.com/download.aspx?id=1051>. [Acceso: 2 - Jun - 2019]

[35] "Aplicaciones de IoT en agricultura", *Agriculturers.com / Red de Especialistas en Agricultura*, 2018. [En línea]. Disponible: <http://agriculturers.com/aplicaciones-de-iot-en-agricultura/>. [Acceso: 23 - Jun - 2019]

- [36] A. MO, "En Peñafiel, Valladolid, hay quienes quieren revolucionar el cultivo de vino con drones", *Xataka.com*, 2018. [En línea]. Disponible: <https://www.xataka.com/especiales/penafiel-valladolid-hay-quienes-quieren-revolucionar-cultivo-vino-drones>. [Acceso: 24 - Jun - 2019]
- [37] B. Remota, "De Silicon Valley a Rosario, con la mirada puesta en el campo - #BorderPeriodismo", *#BorderPeriodismo*, 2018. [En línea]. Disponible: <https://borderperiodismo.com/2018/10/31/de-silicon-valley-a-rosario-con-la-mirada-puesta-en-el-campo/>. [Acceso: 23 - Jun - 2019]
- [38] P. Scully, "The Top 10 IoT Segments in 2018 – based on 1,600 real IoT projects - IoT Analytics", *Iot-analytics.com*, 2018. [En línea]. Disponible: <https://iot-analytics.com/top-10-iot-segments-2018-real-iot-projects/>. [Acceso: 22 - Jun - 2019]
- [39] S. Fourtané, "Reducing Air Pollution in Smart and Sustainable Future Cities", *Interestingengineering.com*, 2018. [En línea]. Disponible: <https://interestingengineering.com/reducing-air-pollution-in-smart-and-sustainable-future-cities>. [Acceso: 22 - Jun - 2019]
- [40] "Qué es Smarthome | Pccomponentes", *Pccomponentes.com*. [En línea]. Disponible: <https://www.pccomponentes.com/que-es-smarthome>. [Acceso: 24 - Jun - 2019]
- [41] "ONP Telecoms - Domótica - IoT internet de las cosas", *ONP Telecoms*. [En línea]. Disponible: <http://www.onptelecoms.com/domotica-iot/>. [Acceso: 04 - Jun - 2019]
- [42] "About - Autopilot", *Autopilot*. [En línea]. Disponible: <https://autopilot-project.eu/about-autopilot/>. [Acceso: 04 - Jun - 2019]
- [43] A. Otero, "El precio del Autopilot de Tesla subirá otros 1.000 dólares, pero las nuevas funciones siguen sin llegar", *Motorpasion.com*, 2019. [En línea]. Disponible: <https://www.motorpasion.com/tesla/precio-autopilot-tesla-subira-otros-1-000-dolares-nuevas-funciones-siguen-llegar>. [Acceso: 04 - Jun - 2019]
- [44] K. Matthews, "Is The Internet of Medical Things (IoMT) on Par With the IoT Market as a Whole? -", *Hitconsultant.net*, 2019. [En línea]. Disponible: https://hitconsultant.net/2019/01/30/is-iomt-tech-iot-market-as-a-whole/#.XTL_fOgzbiU. [Acceso: 22 - Jun - 2019]

- [45] "Internet de las cosas médicas (IoMT): su aplicación | Reporte Digital", *Reporte Digital*, 2018. [En línea]. Disponible: <https://reportedigital.com/iot/internet-de-las-cosas-medicas-iomt/>. [Acceso: 22 - Jun - 2019]
- [46] "Internet industrial de las cosas (IIoT): sus aportes | Reporte Digital", *Reporte Digital*, 2018. [En línea]. Disponible: <https://reportedigital.com/iot/internet-industrial-de-las-cosas-iiot/>. [Acceso: 22 - Jun - 2019]
- [47] "El Internet Industrial de las Cosas transforma los servicios de campo", *Blog.prodware.es*, 2016. [En línea]. Disponible: <https://blog.prodware.es/el-internet-industrial-de-las-cosas-transforma-los-servicios-de-campo/#.XTM5Y-gzblU>. [Acceso: 22 - Jun - 2019]
- [48] "Redes de sensores - IK4-TEKNIKER", *Tekniker.es*. [En línea]. Disponible: <https://www.tekniker.es/es/redes-de-sensores>. [Acceso: 25 - Jun - 2019]
- [49] D. Kumar and T. Kaur, "Computational intelligence-based energy efficient routing protocols with QoS assurance for wireless sensor networks: a survey", *International Journal of Wireless and Mobile Computing*, vol. 16, no. 2, p. 172, 2019 [En línea]. Disponible: <https://www.scopus.com/record/display.uri?eid=2-s2.0-79951581342&origin=inward&txGid=4e3d35e5d823b7ac5de11a258be6fb9e>. [Acceso: 04 - Jun - 2019]
- [50] M. Li, Z. Li and A. Vasilakos, "A Survey on Topology Control in Wireless Sensor Networks: Taxonomy, Comparative Study, and Open Issues", *Scopus*, vol. 101, no. 12, pp. 2538-2557, 2013 [En línea]. Disponible: <https://www.scopus.com/record/display.uri?eid=2-s2.0-84889567969&origin=inward&txGid=4f4d9973e93041cc9025b4993bfb1ee4>. [Acceso: 06 - Jun - 2019]
- [51] M. Younis, I. Senturk, K. Akkaya, S. Lee and F. Senel, "Topology management techniques for tolerating node failures in wireless sensor networks: A survey", *Computer Networks*, vol. 58, pp. 254-283, 2014. [Acceso: 06 - Jun - 2019]
- [52] G. Vijay, E. Ben Ali Bdira and M. Ibnkahla, "Cognition in Wireless Sensor Networks: A Perspective", *IEEE Sensors Journal*, vol. 11, no. 3, pp. 582-592, 2011. [Acceso: 06 - Jun - 2019]

- [53] L. Alazzawi, A. Elkateeb, A. Ramesh and W. Aljuhar, "Scalability Analysis for Wireless Sensor Networks Routing Protocols", *IEEE*, 2008. [Acceso: 06 - Jun - 2019]
- [54] "What is the exact definition of heterogeneity in sensor node?", *Research Gate*. [En línea]. Disponible: https://www.researchgate.net/post/What_is_the_exact_definition_of_heterogeneity_in_sensor_node. [Acceso: 25 - Jun - 2019]
- [55] S. Misra, P. Kar, A. Roy and M. Obaidat, "Existence of dumb nodes in stationary wireless sensor networks", *Journal of Systems and Software*, vol. 91, pp. 135-146, 2014. [Acceso: 06 - Jun - 2019]
- [56] S. Misra and A. Jain, "Policy controlled self-configuration in unattended wireless sensor networks", *Journal of Network and Computer Applications*, vol. 34, no. 5, pp. 1530-1544, 2011. [Acceso: 06 - Jun - 2019]
- [57] E. Karapistoli, A. Economides and P. Sarigiannidis, "SRNET: a real-time, cross-based anomaly detection and visualization system for wireless sensor networks.", *Proceedings of the Tenth Workshop on Visualization for Cyber Security*, pp. 49–56, 2013. [Acceso: 24 - Jun - 2019]
- [58] T. Ojha, S. Misra and N. Raghuwanshi, "Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges", *Computers and Electronics in Agriculture*, vol. 118, pp. 66-84, 2015. [Acceso: 06 - Jun - 2019]
- [59] "Frequently asked questions", *Itu.int*, 2019. [En línea]. Disponible: <https://www.itu.int/net/ITU-R/terrestrial/faq/index.html#g013>. [Acceso: 25 - Jun - 2019]
- [60] "IEEE 802.15.4", *Ieee802.org*, 2019. [En línea]. Disponible: <http://www.ieee802.org/15/pub/TG4.html>. [Acceso: 25 - Jun - 2019]
- [61] *True System-on-Chip solution for 2.4 GHz IEEE 802.15.4 / ZigBee®*. [En línea]. Disponible: <http://www.ti.com/lit/ds/symlink/cc2430.pdf>. [Acceso: 25 - Jun - 2019]
- [62] A. Elahi and A. Gschwender, "2.2 ZigBee Device Types | Introduction to the ZigBee Wireless Sensor and Control Network | InformIT", *Informit.com*, 2009. [En línea]. Disponible: <http://www.informit.com/articles/article.aspx?p=1409785&seqNum=3>. [Acceso: 26 - Jun - 2019]

- [63] "CC2530 Zigbee Module", *Makerfabs*. [En línea]. Disponible: https://makerfabs.com/index.php?route=product/product&product_id=411. [Acceso: 26 - Jun - 2019]
- [64] "What is ZigBee Technology, Architecture and its Applications?", *ElProCus - Electronic Projects for Engineering Students*. [En línea]. Disponible: <https://www.elprocus.com/what-is-zigbee-technology-architecture-and-its-applications/>. [Acceso: 26 - Jun - 2019]
- [65] B. Mitchell, "How Far Will Your Wi-Fi Reach?", *Lifewire*, 2019. [En línea]. Disponible: <https://www.lifewire.com/range-of-typical-wifi-network-816564>. [Acceso: 26 - Jun - 2019]
- [66] "What's the maximum distance a Wi-Fi signal can be reached?", *Quora*, 2019. [En línea]. Disponible: <https://www.quora.com/Whats-the-maximum-distance-a-Wi-Fi-signal-can-be-reached>. [Acceso: 27 - Jun - 2019]
- [67] "Estándar GSM (Sistema global de comunicaciones móviles)", *CCM*. [En línea]. Disponible: <https://es.ccm.net/contents/681-estandar-gsm-sistema-global-de-comunicaciones-moviles>. [Acceso: 27 - Jun - 2019]
- [68] J. Huidobro Moya, "Comunicaciones Móviles. Sistemas GSM, UMTS y LTE", *Google Books*, 2019. [En línea]. Disponible: <https://books.google.es/books?id=go2fDwAAQBAJ&printsec=frontcover&dq=gsm&hl=es&sa=X&ved=0ahUKEwjju9mryMDjAhVOi1wKHayzCAMQ6AEIKzAA#v=onepage&q=gsm&f=false>. [Acceso: 27 - Jun - 2019]
- [69] V. Granatstein, *Physical Principles of Wireless Communications*, 2nd ed. Silver Spring, Maryland, 2007, p. 29. [Acceso: 25 - Jul - 2019]
- [70] "¿Qué es GSM y cómo funciona? | VIU", *Universidadviu.es*. [En línea]. Disponible: <https://www.universidadviu.es/que-es-gsm-y-como-funciona/>. [Acceso: 27 - Jun - 2019]
- [71] "Velocidades de Internet móvil 3G y 4G de cada operadora en España", *Testdevelocidad.es*. [En línea]. Disponible: <https://www.testdevelocidad.es/velocidad-internet-movil-3g-4g/>. [Acceso: 02 - Jul - 2019]
- [72] A. Hevia, "Cómo localizar mi posición: GPS, A-GPS, WIFI y redes GSM", *Xatakamovil.com*, 2011. [En línea]. Disponible:

<https://www.xatakamovil.com/conectividad/como-localizar-mi-posicion-gps-a-gps-wifi-y-redes-gsm>. [Acceso: 28 - Jun - 2019]

[73] "GSM localizer without GPS – Part 1 Introduction", *Open Electronics*, 2010. [En línea]. Disponible: <https://www.open-electronics.org/gsm-localizer-without-gps-part-1-introduction/>. [Acceso: 28 - Jun - 2019]

[74] "OpenCelliD - Largest Open Database of Cell Towers & Geolocation - by Unwired Labs", *Opencellid.org*. [En línea]. Disponible: <http://www.opencellid.org>. [Acceso: 28 - Jun - 2019]

[75] "Raspberry Pi Location Tracker", *Instructables*, 2017. [En línea]. Disponible: <https://www.instructables.com/id/Raspberry-Pi-Location-Tracker/>, <https://antenasgsm.com/39.8311807928473/-3.1802600639999845/12/true,true,true,true,true>. [Acceso: 28 - Jun - 2019]

[76] "Posicionamiento (Localización)", *movilfacil*, 2011. [En línea]. Disponible: <https://movilfacil.wordpress.com/2011/03/25/cap-8-posicionamiento-localizacion/>. [Acceso: 28 - Jun - 2019]

[77] "What is a Raspberry Pi?", *Raspberry Pi*. [En línea]. Disponible: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>. [Acceso: 08 - Ago - 2018]

[78] "TOP 100 de proyectos con Raspberry pi", *Soloelectronicos*, 2019. [En línea]. Disponible: <http://soloelectronicos.com/2017/04/24/top-100-de-proyectos-con-raspberry-pi/>. [Acceso: 08 - Ago - 2018]

[79] "Qué es un sensor", *Profesormolina.com.ar*, 2019. [En línea]. Disponible: http://www.profesormolina.com.ar/tecnologia/sens_transduct/que_es.htm. [Acceso: 10 - May - 2019]

[80] "Sensor", *Es.wikipedia.org*. [En línea]. Disponible: <https://es.wikipedia.org/wiki/Sensor>. [Acceso: 10 - May - 2019]

[81] "Raspbian", *Es.wikipedia.org*. [En línea]. Disponible: <https://es.wikipedia.org/wiki/Raspbian>. [Acceso: 10 - May - 2019]

[82] "FrontPage - Raspbian", *Raspbian.org*. [En línea]. Disponible: <https://www.raspbian.org/>. [Acceso: 10 - May - 2019]

- [83] "Python", *Es.wikipedia.org*. [En línea]. Disponible: <https://es.wikipedia.org/wiki/Python>. [Acceso: 10 - May - 2019]
- [84] "BeginnersGuide/Programmers - Python Wiki", *Wiki.python.org*. [En línea]. Disponible: <https://wiki.python.org/moin/BeginnersGuide/Programmers>. [Acceso: 10 - May - 2019]
- [85] "Global Cellular Network for IoT & M2M Connectivity", *Hologram*. [En línea]. Disponible: <https://hologram.io/products/cellular/>. [Acceso: 10 - Jul - 2019]
- [86] "Learn More - ThingSpeak IoT", *Thingspeak.com*. [En línea]. Disponible: https://thingspeak.com/pages/learn_more. [Acceso: 10 - Jul - 2019]
- [87] "R: What is R?", *R-project.org*. [En línea]. Disponible: <https://www.r-project.org/about.html>. [Acceso: 10 - Feb - 2019]
- [88] "About us", *draw.io*. [En línea]. Disponible: <https://about.draw.io/about-us/>. [Acceso: 02- Sep- 2019]
- [89] "IBM Knowledge Center", *Ibm.com*. [En línea]. Disponible: https://www.ibm.com/support/knowledgecenter/en/SSB27H_6.2.0/fa2ti_what_is_socket_connection.html. [Acceso: 10 - May - 2019]
- [90] "Servicio de mensajes cortos", *Es.wikipedia.org*. [En línea]. Disponible: https://es.wikipedia.org/wiki/Servicio_de_mensajes_cortos. [Acceso: 10 - Jul - 2019]
- [91] "Correo electrónico", *Es.wikipedia.org*. [En línea]. Disponible: https://es.wikipedia.org/wiki/Correo_electr%C3%B3nico. [Acceso: 10 - Jul - 2019]
- [92] "SSH (Secure Shell) Home Page | SSH.COM", *Ssh.com*. [En línea]. Disponible: <https://www.ssh.com/ssh/>. [Acceso: 10 - Jul - 2019]
- [93] "Plantilla Especificación de Requisitos | Marco de Desarrollo de la Junta de Andalucía", *Juntadeandalucia.es*. [En línea]. Disponible: <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/456>. [Acceso: 07 - Ago - 2019]
- [94] "Guía para la redacción de casos de uso | Marco de Desarrollo de la Junta de Andalucía", *Juntadeandalucia.es*. [En línea]. Disponible: <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416>. [Acceso: 07 - Ago - 2019]

- [95] "BOE.es - Documento BOE-A-2017-2536", *Boe.es*, 2017. [En línea]. Disponible: <https://www.boe.es/buscar/doc.php?id=BOE-A-2017-2536>. [Acceso: 14 - Ago - 2019]
- [96] "BOE.es - Documento consolidado BOE-A-2018-16673", *Boe.es*, 2018. [En línea]. Disponible: <https://www.boe.es/buscar/act.php?id=BOE-A-2018-16673>. [Acceso: 14 - Ago - 2019]
- [97] "EUR-Lex - 32016R0679 - EN - EUR-Lex", *Eur-lex.europa.eu*, 2016. [En línea]. Disponible: <https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1552662547490&uri=CELEX%3A32016R0679>. [Acceso: 14 - Ago - 2019]
- [98] "DECISIÓN DE EJECUCIÓN (UE) 2016/1250 DE LA COMISIÓN de 12 de julio de 2016 con arreglo a la Directiva 95/46/CE del Parlamento Europeo y del Consejo, sobre la adecuación de la protección conferida por el Escudo de la privacidad UE-EE.UU.", *Eur-lex.europa.eu*, 2016. [En línea]. Disponible: <https://eur-lex.europa.eu/legal-content/ES/TXT/PDF/?uri=CELEX:32016D1250&from=EN>. [Acceso: 15 - Ago - 2019]
- [99] "Privacy Policy", *Hologram*. [En línea]. Disponible: <https://hologram.io/privacy-policy/>. [Acceso: 07 - Jul - 2019]
- [100] "BOE.es - Documento BOE-A-2014-13259", *Boe.es*, 2014. [En línea]. Disponible: <https://www.boe.es/buscar/doc.php?id=BOE-A-2014-13259>. [Acceso: 14 - Ago - 2019]
- [101] "Módem", *Es.wikipedia.org*. [En línea]. Disponible: <https://es.wikipedia.org/wiki/Modem>. [Acceso: 11 - May - 2019]
- [102] "What Is Big Data? - Gartner IT Glossary - Big Data", *Gartner IT Glossary*. [En línea]. Disponible: <https://www.gartner.com/it-glossary/big-data/>. [Acceso: 07 - May - 2019]
- [103] "Unix", *Es.wikipedia.org*. [En línea]. Disponible: <https://es.wikipedia.org/wiki/Unix>. [Acceso: 11 - May - 2019]
- [104] M. Sirvent, "Nuevas plantaciones vid: ¿Vaso o Espaldera? Ventajas e inconvenientes.", *VitiViniCultura.net: Arte y técnica de cultivar vid y elaborar vino*, 2016. [En línea]. Disponible: <http://www.vitivinicultura.net/vaso-o-espaldera.html>. [Acceso: 17 - Ene - 2019]
- [105] A. Avilés Pozo, "Cepas contra ZEPAs: sigue "sin soluciones" el conflicto por las viñas en espaldera", *eldiario.es*, 2016. [En línea]. Disponible:

https://www.eldiario.es/clm/Cepas-ZEPAs-soluciones-conflicto-espaldera_0_546645512.html. [Acceso: 14 - Abr - 2019]

[106] J. Togores, *La calidad del vino desde el viñedo*. Mundi-Prensa, 2006, pp. 255-260. [Acceso: 14 - Abr - 2019]

[107] "Riego automático en viña 'Vitis vinifera' cv. Tempranillo en función de los niveles de humedad del suelo", *Interempresas*, 2013. [En línea]. Disponible: <http://www.interempresas.net/Horticola/Articulos/117360-Riego-automatico-vina-Vitis-vinifera-cv-Tempranillo-funcion-niveles-humedad-suelo.html>. [Acceso: 14 - Feb - 2019]

[108] G. Hargreaves and R. Allen, "History and Evaluation of Hargreaves Evapotranspiration Equation", *Journal of Irrigation and Drainage Engineering*, vol. 129, no. 1, pp. 53-63, 2003. [Acceso: 06 - Jun - 2019]

[109] Z. Samani, "Discussion of "History and Evaluation of Hargreaves Evapotranspiration Equation" by George H. Hargreaves and Richard G. Allen", *Journal of Irrigation and Drainage Engineering*, vol. 130, no. 5, pp. 447-448, 2004. [Acceso: 06 - Jun - 2019]

[110] "Agencia Estatal de Meteorología - AEMET. Gobierno de España", *Aemet.es*. [En línea]. Disponible: <http://www.aemet.es/es/portada>. [Acceso: 07 - Feb - 2019]

[111] J. Honaker, G. King and M. Blackwell, "Amelia II: A Program for Missing Data", *Journal of Statistical Software*, vol. 45, no. 7, 2011.

[112] A. Rivera, "Las nubes absorben más radiación solar de lo esperado", *EL PAÍS*, 1995. [En línea]. Disponible: https://elpais.com/diario/1995/02/01/sociedad/791593214_850215.html. [Acceso: 07 - Ago - 2019]

[113] L. De Stefano, "El uso y gestión de las aguas subterráneas - Guía Nueva Cultura del Agua", *Fnca.eu*. [En línea]. Disponible: <https://fnca.eu/guia-nueva-cultura-del-agua/la-planificacion-y-gestion-del-agua-en-espana/el-uso-y-gestion-de-las-aguas-subterraneas?&imprimir=1>. [Acceso: 07 - Ago - 2019]

[114] "¿Qué es un acuífero? Todo sobre aguas subterráneas | Ingeoexpert", *Ingeoexpert*, 2018. [En línea]. Disponible: <https://ingeoexpert.com/acuifero-aguas-subterraneas/?v=7516fd43adaa>. [Acceso: 07 - Ago - 2019]

[115] R. Allen, L. Pereira, M. Smith, D. Raes and J. Wright, "FAO-56 Dual Crop Coefficient Method for Estimating Evaporation from Soil and Application Extensions", *Journal of Irrigation and Drainage Engineering*, vol. 131, no. 1, pp. 2-13, 2005. [Acceso: 06 - Jun - 2019]

Anexos

Anexo 1. Summary

In this annex, there is a summary of the whole project developed in English, as it is a requirement for this document to be valid.

1.1. Introduction

This project is intended to cover the concepts of Internet of Things and Wireless Sensor Networks. In order to do this, this document has a solution involving the last two concepts for two big issues related to agriculture: farming irrigation and agricultural machines' monitoring.

On the one hand, irrigation is a crucial method for keeping plants alive in dry seasons. However, due to global warming, the weather is changing irregularly, and some areas are getting drier and warmer each year. Without water, the quality of the fruit is low, and the plants' average lifespan is shorter because they are exposed to continuous bad conditions.

To avoid this problem, people use groundwater from aquifers. In order to get this, they need to use diesel engines [6] or solar panels to give energy to water pumps (Figure 1.1 and Figure 1.2. respectively). In terms of efficiency, many people prefer diesel engines

because they can work at night without the energy of the Sun. However, these engines are quite valuable and many of them have been stolen in the last 10 years [8] because they are unprotected in farmers' lands.

1.2. Tools and devices

In the following section, plenty of tools and devices used in this project will be described as they are going to be divided into two groups: Software and Hardware.

1.2.1. Software

The software tools used in this project are the following:

- ❖ **Python.** The main programming language in the project due to its usefulness and simplicity to get information from the sensors and analyze it.
- ❖ **Raspbian.** The official operating system of the Raspberry Pi.
- ❖ **Microsoft Office 365 – Microsoft Word.** Used for the project documentation.
- ❖ **Microsoft Office 365 – Microsoft Excel.** Used for creating Gantt charts.
- ❖ **MobaXterm.** It is the terminal in which SSH connections can be made from the computer to the Raspberry Pi.
- ❖ **RStudio.** It is the integrated development environment for R programming that was used for data gathering and processing in Annex 2.
- ❖ **R.** Programming language used for the data in Annex 2.
- ❖ **Hologram Cloud.** This is the place where all the data arrive from Raspberry Pi. They are handled there by the administrator and SMS and alerts are sent to the client via SMS and emails.
- ❖ **Thingspeak.** Web page that displays data from the moisture sensor in real time.
- ❖ **Draw.io.** Used for creating flowcharts and other diagrams in the document.

1.2.2. Hardware

In terms of Hardware, the devices used in this project are the following:

- ❖ **Raspberry Pi 3 Model B.** As a small computer, Raspberry Pi has several ports: USB, HDMI, Ethernet, Audio Jack, microSD, touchscreen module, camera module, Bluetooth and WiFi. Quad Core 1.2GHz ARM 64-bit CPU with 1 GB of RAM. Furthermore, it has 40 GPIO pins used for the sensors as we can see in Figure 1.3.
- ❖ **FC-28 moisture sensor.** It measures the amount of humidity in the ground by computing the conductivity variation. Widely used in automatic irrigation systems in order to activate the pump system when it is necessary (Figure 1.4). It has two outputs: analog, where values go from 0 (underwater) to 1023 (dry), and digital that gives a HIGH value when the humidity reaches a threshold [10].
- ❖ **MCP3008.** It is a 10-bit analog to digital converter which is designed and implemented by Microchip [11]. It is useful in this project because Raspberry Pi does not have analog pins. Then, with MCP3008, the Raspberry Pi would be able to read all the exact values from the FC-28 (Figure 1.6 and Figure 1.7).
- ❖ **HC-SR501.** It is a passive infrared sensor which detects infrared radiation variations in the environment such as the body heat from humans or animals. Its sensor aperture is of 110° and its detection distance can be calibrated with a potentiometer from 3 to 7 meters (Figure 1.8 and Figure 1.9).
- ❖ **MPU6050.** It is a 6-axis gyroscope (3-axis to watch the speed and 3-axis for the rotation). It uses I2C communication, one for the synchronized clock (SCL) and other for the data (SDA) (Figure 1.11).
- ❖ **Hologram SIM.** It is a secure IoT SIM card for international communications. It is easily activated by Hologram and it is not necessary to talk to local telephone companies since Hologram has already done this task. This SIM card uses the telephone network which has more power (Figure 1.12).
- ❖ **Hologram Nova.** This is the recommended modem by Hologram in order to have an Internet connection in any device. It is simple to use since the modem is plugged into the Raspberry Pi and the Hologram python library is installed there.
- ❖ **Power bank.** In order to keep the Raspberry Pi on many kilometers away from home, it is necessary to have a power bank. In this project, a battery of 10000 mAh is used and it lasts for 12 hours approximately.

1.3. Objectives

The aim of this project is to design and implement a monitoring system for farming irrigation (specially in vineyards) and agriculture equipment (diesel engines), by using wireless sensors that are able to send data to the system in order to analyze and make decisions by its own. The two main objectives are the following:

- ❖ Avoiding the waste of water for an optimal irrigation.
- ❖ Avoiding robbery of diesel engines by using a security protocol.

To fulfill with the objectives, this project will use the sensor monitoring context to design a system with three possible scenarios:

In the first scenario, the Raspberry Pi will send all the humidity data to a web page. When it gets a new humidity value, it automatically sends the data to the web page. There, all the data are displayed in a plot versus time where the client can see the humidity evolution during time and download it in different formats. However, the user needs permissions from the administrator to see the activity in the web page.

The second scenario consists of a terminal connected to the Raspberry Pi and the sensors where the administrator can access to everything such as starting or killing processes, opening the log files, shutting down or rebooting the system.

In the third scenario, the client and the administrator interact with the server. While the client receives alerts from the system via SMS and email, the administrator handles the routing information to the client and other possible issues. The server is an intermediate stage between the sensors and the users.

The general system architecture is the following one:

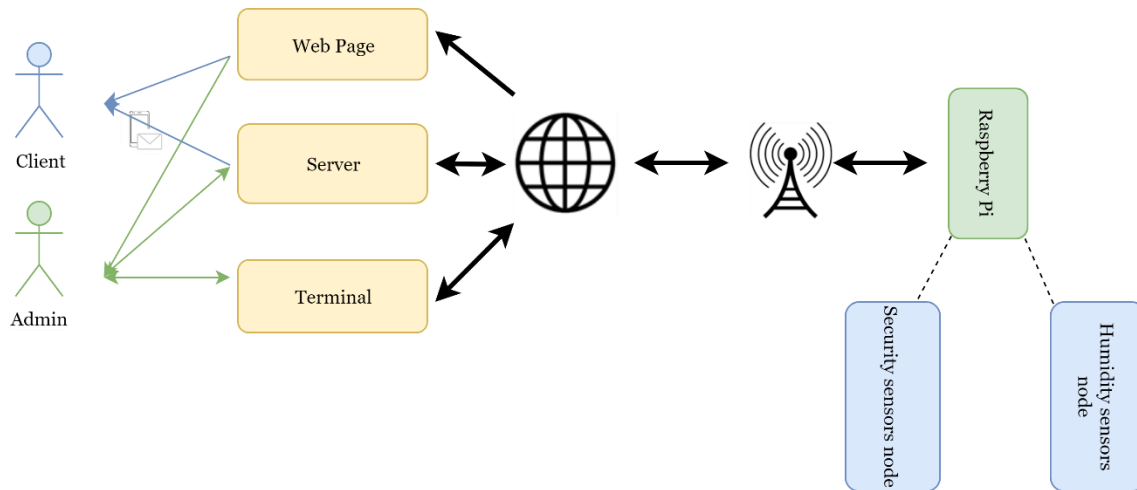


Figure A.1.1: Architecture of the system

1.4. Implementation

In this section, we are going to see the different steps in order to make this project possible. There are two main programs: `riego.py` and `seguridad.py` and they monitor the irrigation system and the diesel engine respectively.

1.4.1. Riego.py

In this program, there are several objectives: get the humidity value from the moisture sensor, send it to the web page (to see the temporal plot), start and stop the diesel engine and create a backup for all the moisture values in a log file. All this information is summarized in figure 4.1.

On the other hand, there is also an auxiliary program called `mcp3008.py` which is helpful to get the data from the sensor because of the lack of analog pin in the Raspberry Pi.

1.4.1.1. Humidity analysis

According to the flowchart in Figure 4.1, there are three conditions to turn on or off the diesel engine:

- ❖ Humidity greater than 50 %. The engine is off.
- ❖ Humidity lower than 20 %. The engine is on until it reaches 30%

- ❖ Humidity between 20% and 50%. If the level of water for this day is reached, then the engine stops.

The third condition points to the concept of **water available each day**. For these plants, it is important to keep a good level of humidity but also to irrigate more in different periods of the year. For example: when the grapes are growing and ripening (see Annex 2 for more information about vineyards). Furthermore, it is important to consider the useful rain because the more it rains, the less ground water the vineyards need. In an illustrative way, we will use Table A.2.4 as a reference to know how many liters of water per day we need each month.

To compute the **water consumption** now, we will use the datasheet from the diesel engine [2]. In normal conditions, with a water pump, this diesel engine [2] extracts 8300 liters per hectare and hour or $2,3055 \cdot 10^{-4}$ liters per square meter and second (887,32 gallons per acre and hour). Therefore, to compute the water consumption, we would use the following computation:

$$Water_{Consumption} \left(\frac{L}{m^2} \right) = Rate_{Extraction} \left(\frac{L}{m^2 s} \right) \cdot Time_{ActiveEngine} (s)$$

1.4.1.2. Sending data to ThingSpeak

When the moisture sensor takes data from the ground, the program will send all the data automatically to ThingSpeak. In order to do this, the program will connect first to the platform by using the ThingSpeak write API key (Figure 4.4). Then, when the Raspberry Pi is connected, it will send data via POST request as we can see in Figure 4.4. Finally, the data is displayed in a time plot and the user can download the values via CSV, XML or JSON (Figure 4.6).

1.4.1.3. Log_riego.txt

For the sake of prevention, in case there is any issue with the program performance, the administrator can check the humidity values without using ThingSpeak. This file is stored in the Raspberry Pi and the information will be added in the format: <Date> <Water consumption> <Humidity> <time_active> <time_stopped> (Figure 4.7).

1.4.2. Seguridad.py

This program will determine if there is a suspicious activity (robbery) of the diesel machine. In order to do this, it will check if there is a variation of infrared radiation in the area (a person in movement) and the rotation and speed of the gyroscope (engine movement). When the parameters show a suspicious activity, the user will be alerted, and the program will send the engine location by using a Google Maps link. This activity is described in Figure 4.8.

1.4.2.1. System activation and deactivation

Sometimes, it is necessary for the farmer to stop the engine because he or she needs to refill the diesel tank. On the other hand, it is also necessary to log into the system in order to send the activation or deactivation command because anyone could send messages to the device.

Hologram offers two options to send SMS to the device: via Hologram Cloud, which is a free service, or acquiring a new telephone number for the Hologram SIM card (for 1 dollar a month) and send SMS to it. Despite SMS via Hologram Cloud are free, this service is operated by the administrator so the second option should be better because the client activates and deactivates the service when he or she wants.

In seguridad.py, there will be some functions that will handle the SMS processing. At the beginning of the program, it will send to the client's phone an SMS with a random code. If the client replicates the same code, then the security program will be activated. Then, the program will check if the user wants to activate or deactivate the security process by receiving either A (activate) or D (deactivate).

1.4.2.2. PIR sensor

As explained before, it will check if there is infrared radiation variation in the detection area. Furthermore, to avoid false alarms due to small animals, the best solution is to place the sensor at a height of 1.5 meters because the line of detection would not find a small animal.

1.4.2.3. Gyroscope

This is an additional resource in case the PIR sensor does not detect anyone. With this sensor, we will analyze the rotation and the linear speed movement of the diesel engine. First, we will assume that the gyroscope is placed in horizontal (Figure 4.11). Then, in terms of rotation, considering that z-axis does not give real values because it needs another gyroscope to have the reference, x-axis and y-axis will be enough to analyze rotation. Regarding the speed, z-axis will be 1g all the time (g is the Earth gravitational force, or 9.8 m/s^2 ; see Figure 4.12) so its value is useless as well.

On the other hand, after testing the sensor (Figure 4.10), the conditions to check suspicious robbery are:

- ❖ Rotation: $|x\text{-axis}| > 8^\circ$ or $|y\text{-axis}| > 8^\circ$
- ❖ Speed: $|x\text{-axis}| > 0.4g$ or $|y\text{-axis}| > 0.4g$

1.4.2.4. Device location

When the user is alerted, the device (Raspberry Pi) will send its location by using GSM triangulation (get the location using the cell towers nearby [72, 73, 74, 75]). In this case, it is necessary to have Hologram Nova plugged into the Raspberry Pi (Figure 4.16).

1.4.2.5. Sending alerts to the user

As it is said before, the user is alerted via SMS and email. In order to do this, the program sends the alert to Hologram Cloud and then, these alerts are routed to the client. Routing works with tags or topics and for example: when there is a suspicious activity related to the PIR sensor, then, it will send the message to the cloud including “PIR_SENSOR” as tag. The cloud will classify all the messages with “PIR_SENSOR” and send the message to the user (same with the other tags, see Figure 4.17, Figure 4.18, Figure 4.19, Figure 4.20).

1.4.2.6. Log_seguridad.txt and alertas.txt

For the same purpose as `log_riego.txt`, **log_seguridad.txt** will save all the information related to the speed and the rotation (gyroscope) and **alertas.txt** will

write the alert that is triggering (either the PIR sensor or the gyroscope, Figure 4.21 and Figure 4.22).

1.4.3. Terminal

This is the third way to access the data. There will be an SSH connection between our computer and the Raspberry Pi. Since it is not easy to get the modem public IP address, we used Spacebridge, a Hologram service that is used for SSH connections. It creates a tunnel between our computer and the Raspberry Pi using the API key from Hologram and the chosen ports to create the connection (in this case, port 12345 was used for the computer and port 22 for the Raspberry Pi). Once the tunnel is created, the user can SSH to the local IP address given by the Spacebridge program. The administrator can log into the Raspberry Pi and check the log files to see the results, stop or start processes or shutdown and rebooting the Raspberry Pi.

1.4.4. Data consumption

The modem data consumption is free unless it reaches 1 MB each month. To check the consumption, the administrator must log into the Cloud and click on 'Usage' (Annex 3).

1.4.5. All in one

To run `riego.py`, `seguridad.py` and to start the Internet connection (`sudo hologram network connect`) with Hologram Nova, `crontab` is used. It will run the three tasks at the same time when the Raspberry is booted up in the following way:

```
@reboot sudo hologram network connect && sleep 10 && sudo  
python2 /home/pi/Desktop/Hologram/seguridad.py && sleep 10  
&& sudo python2 /home/pi/Desktop/Hologram/riego.py
```

To make sure about the functionality of the process, there is an extra 10 seconds delay between tasks.

1.5. Testing

All the system requisites were fulfilled in Chapter 5, except the communication via SMS in Spain. Due to Hologram limitations, a Spanish phone could not receive or send SMS to the device. However, it works with networks from the United States. Therefore, we tried to send alerts to an American phone, and it worked as we can see in Figure 4.33.

1.6. Conclusion

Regarding the project itself, it required a good level of knowledge of different technologies. The idea was well defined from the beginning and now, it is a functional IoT project.

Finally, despite all the efforts to make this project possible, I am happy with the result. I must thank all the help from my mentor and from all my friends throughout the project.

Anexo 2. Ampliación de conceptos sobre la vid

La economía principal de Corral de Almaguer y de los municipios de la zona es el vino. Esta bebida se obtiene de la uva mediante un proceso llamado fermentación alcohólica de su mosto o zumo.

La uva se produce en las cepas de vid. Son arbustos que tienen un tamaño aproximado entre 30 cm y 1 m y dependiendo de la variedad de la uva tienen características diferentes como el tamaño de la uva, su forma, su calidad (nivelado en grados alcohólicos), entre otras... Además, se pueden distinguir infinidad de variedades de uva como el Tempranillo, Airén, Cabernet Sauvignon, Garnacha o Moscatel.

Por otro lado, las viñas pueden ser en vaso o en espaldera. Las primeras son más naturales y requieren de un menor coste de implementación y mantenimiento que el sistema en espaldera, pero la recolección de la uva se hace en un modo manual y se precisan de más días. Por otro lado, el sistema en espaldera es más costoso ya que se necesita un sistema de barras y cables de acero inoxidable para poder sujetar las cepas y hacer que estas crezcan hacia arriba (no a los lados como las cepas en vaso) para que la máquina pueda recoger la uva más fácilmente. Sin embargo, la recolección se hace más rápida y efectiva.



Figura A.2.1: Cepas en vaso [104]



Figura A.2.2: Cepas en Espaldera [105]

2.1. El ciclo vegetativo de la vid

La vid es de hoja caduca y vive todo un ciclo vegetativo y anual en el que se desarrollan una serie de procesos que dan lugar a la uva final.

2.1.1. Brotación

En esta fase, se produce el nacimiento de la vid. Ocurre en marzo, a comienzos de la primavera donde se pueden observar los primeros brotes que crecerán hasta convertirse en hojas.

2.1.2. Foliación

En esta etapa, aparecen las primeras hojas de la vid. Además, se producen los primeros azúcares y ácidos en las mismas que condicionarán la calidad de la uva.

2.1.3. Floración

También se conoce como el momento en el que aparecen los embriones de las flores (los granos de las uvas que formarán los racimos). Suele ser a finales de mayo y principios de junio.

2.1.4. Fecundación y fructificación

A principios de junio hasta principios de julio, las flores empiezan a dar frutos. Estos todavía no están listos para su recolección, hay que esperar a su maduración.

2.1.5. Envero y maduración

Esta etapa se desarrolla a lo largo del verano (hasta finales de septiembre) y en ese momento la uva va cambiando de color (en función de su variedad) hasta alcanzar su tonalidad final.

2.1.6. Vendimia y parada

A finales de septiembre y a principios de octubre se lleva a cabo la vendimia o la recolección de la uva. Cuando termina, la vid entra en un proceso de parada vegetativa donde esta “duerme” desde noviembre hasta marzo.

2.2. Necesidad hidrológica

Las cepas de vid necesitan el agua suficiente para poder crecer, tener fruto y poder mantenerlo. De acuerdo con José Ramón Lissarrague [106], la vid es capaz de sobrevivir entre niveles de pluviometría de 300 a 800 mm anuales.

Además, es necesario tener en cuenta el ciclo vegetativo de la vid ya que la planta necesita más o menos agua en función de la fase en la que se encuentre. Asumiendo que 400 mm es suficiente para las plantas, el reparto de agua en las diferentes etapas sería la siguiente según los porcentajes de Lissarrague:

- Parada o periodo invernal 2 % (8 mm)
- Brotación y foliación 10 % (40 mm)
- Fecundación y fructificación 43 % (172 mm)
- Envero y maduración 45 % (180 mm)

Por otro lado, de acuerdo con un grupo de investigación (Oncins, Lampreave, Nadal, Doix [107]), también se han de mantener unos niveles de humedad en el suelo entre el 20% y el 50%.

2.3. Índice de Precipitación

En Corral de Almaguer, el índice de precipitación se considera bajo. No es óptimo y a medida que van pasando los años, generalmente, las lluvias cada vez son más escasas. Por otro lado, también es cierto que hay años buenos, pero no muy frecuentes. Esto supone un problema difícil de resolver porque sin la suficiente agua, no se conseguirá llegar al objetivo final (alrededor de unos 350 mm anuales).

2.3.1. Evapotranspiración

Es necesario considerar que no toda la lluvia es agua útil para las plantas, sino que también se evapora. La evapotranspiración depende de la época del año y también su resultado depende de las temperaturas y de la radiación del sol. Para calcular la evapotranspiración, se ha de usar la fórmula de Hargreaves [108]:

$$ET_o = 0.0135 \cdot (t_{med} + 17.78) \cdot R_s$$

Donde:

- ET_o : Evapotranspiración potencial diaria, mm/día.
- t_{med} : Temperatura media, °C
- R_s : Radiación solar incidente, mm/día.

La radiación solar incidente se calcula a partir de la radiación solar extraterrestre (Es aquella que iría de la parte exterior de la atmósfera al suelo si esta no existiese). Para ello la fórmula de Samani [109] será útil en este caso:

$$R_s = 0.17 \cdot R_0 \cdot (t_{max} - t_{min})^{0.5}$$

Donde:

- R_0 : Radiación extraterrestre. Para calcular su valor, es necesario usar la tabla de Allen (Anexo 4) en función de la latitud y la temperatura.
- t_{max} : Temperatura máxima del día, °C
- t_{min} : Temperatura mínima del día, °C

Es interesante saber cuál es la evapotranspiración diaria en Corral de Almaguer ya que sería muy efectivo para calcular futuros parámetros de control en el riego. Usando la tabla del anexo A, las coordenadas de la localidad son 39° 45', por lo tanto, al encontrarse más cercano al valor de 40°, se optará a recoger esos valores para los cálculos:

Latitud	Radiación extraterrestre (hemisferio norte)											
40	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dec
	6.1	8.3	11.1	14.2	16.2	17.1	16.7	15.0	12.2	9.2	6.7	5.6

Tabla A.2.1: Radiación extraterrestre a 40° de latitud

Una vez se tiene la radiación extraterrestre, es necesario averiguar la temperatura máxima y mínima. Para ello, se utilizará la base de datos de la Agencia Estatal de Meteorología (AEMET) [110]. Los datos vienen en un archivo “.json” en el siguiente formato:

```

1  [ {
2    "fecha" : "2017-01-01",
3    "indicativo" : "4061X",
4    "nombre" : "QUINTANAR DE LA ORDEN",
5    "provincia" : "TOLEDO",
6    "altitud" : "691",
7    "tmed" : "-0,8",
8    "prec" : "0,0",
9    "tmin" : "-1,9",
10   "horatmin" : "04:30",
11   "tmax" : "0,4",
12   "horatmax" : "14:50",
13   "dir" : "21",
14   "velmedia" : "1,1",
15   "racha" : "3,6",
16   "horaracha" : "12:20",
17   "presMax" : "947,3",
18   "horaPresMax" : "00",
19   "presMin" : "940,8",

```

Tabla A.2.2: Datos extraídos de AEMET

Para nuestra conveniencia, se necesita hacer limpieza y procesamiento de los datos ya que hay muchos parámetros que no son necesarios para los futuros cálculos. También, es necesario mencionar que los datos provienen de la estación de Quintanar de la Orden (Toledo), la cual es la estación de AEMET más cercana a Corral de Almaguer (20 kilómetros de distancia).

Como se verá en el siguiente gráfico y en los siguientes, se han recogido datos por campañas o temporadas ya que el ciclo vegetativo de la viña empieza en noviembre y acaba en octubre del año siguiente. En ese caso, se van a utilizar datos de la campaña de 2016 – 2017 ya que faltan datos en el año 2018 y no se pueden estimar correctamente los datos perdidos.

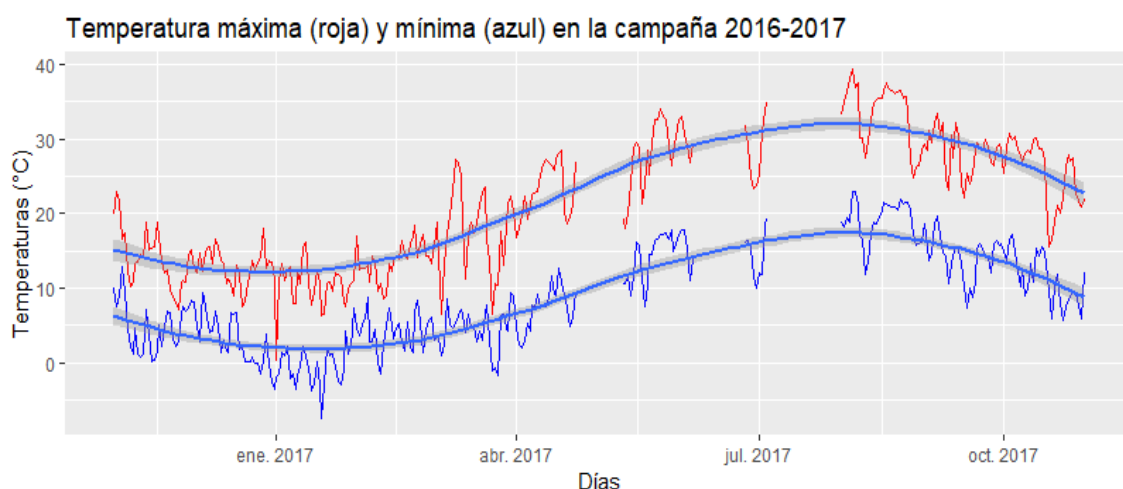


Figura A.2.3: Temperaturas máximas y mínimas en la campaña 2016 – 2017

Sin embargo, también se puede apreciar que hay datos que faltan en los meses de verano y mayo. Para tener una observación aproximada, se ha aplicado regresión (la línea azul en el gráfico). Como se puede observar, los máximos en las variables de temperatura máxima y mínima se encuentran a finales de julio y principios de agosto de 2017 (como se puede esperar).

Con los datos obtenidos anteriormente se puede determinar cuál es la radiación solar (en mililitros día). Para ello, se utiliza la fórmula de Samani.

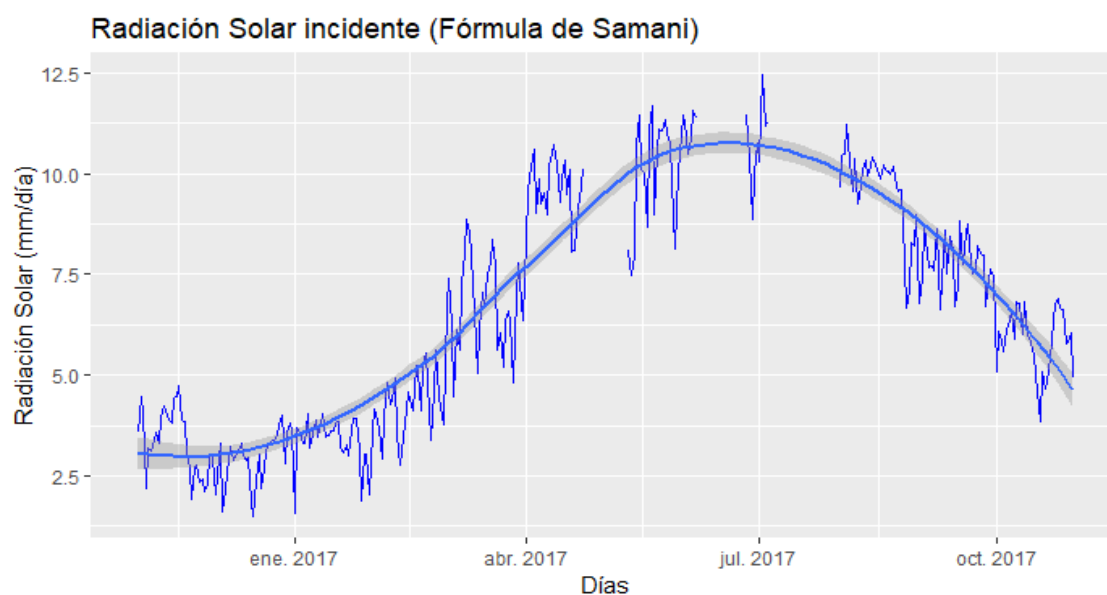


Figura A.2.4: Radiación solar incidente (Fórmula de Samani)

Nótese que al igual que la Figura A.2.2, se puede apreciar que faltan algunos datos en los meses de verano y en mayo. Se ha realizado la misma regresión que antes (Línea curva azul) para ver su tendencia en los lugares donde no tenemos datos.

Una vez que se consigue la radiación solar, se puede calcular la evapotranspiración diaria. En este caso, se necesita el parámetro de temperatura media.

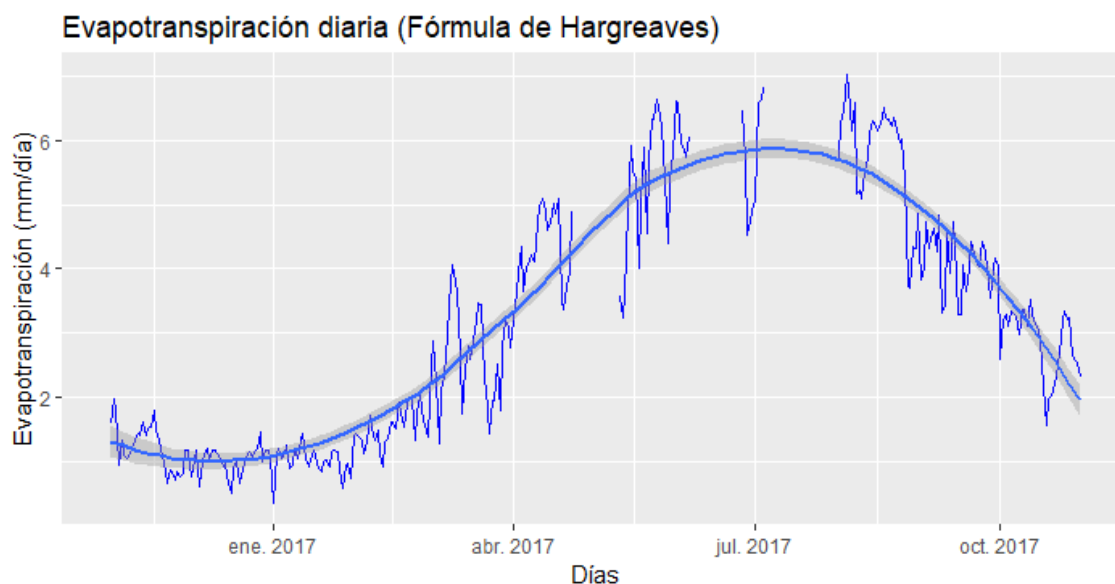


Figura A.2.5: Evapotranspiración diaria

Se puede observar que en los días de verano de 2017 la evapotranspiración puede llegar hasta 6 litros por metro cuadrado (La pérdida es potencial ya que para que se produzca una pérdida del 100% estimado, debe haber agua en la capa más superficial de la tierra). Con esto se puede justificar que las plantas necesitan más agua en verano. Además, durante los periodos de fecundación y fructificación y envero y maduración, la planta necesita más agua. Por lo tanto, ante la necesidad mayor de agua en estos periodos y la mayor evapotranspiración, es crucial optimizar el riego.

También, es necesario tener en cuenta que los datos obtenidos se consiguen en la capa más superficial de la tierra ya que las capas interiores están “más protegidas” por las más superficiales. Por lo tanto, la evapotranspiración se reducirá a medida que se va avanzando en las capas.

2.4. Precipitación en la temporada 2016 - 2017

Una vez se ha calculado la evapotranspiración diaria, conviene calcular cuál es la cantidad de agua de lluvia que se pierde. Para ello, se vuelve a usar la misma base de datos de AEMET para poder calcular la precipitación en la campaña 2016 - 2017.

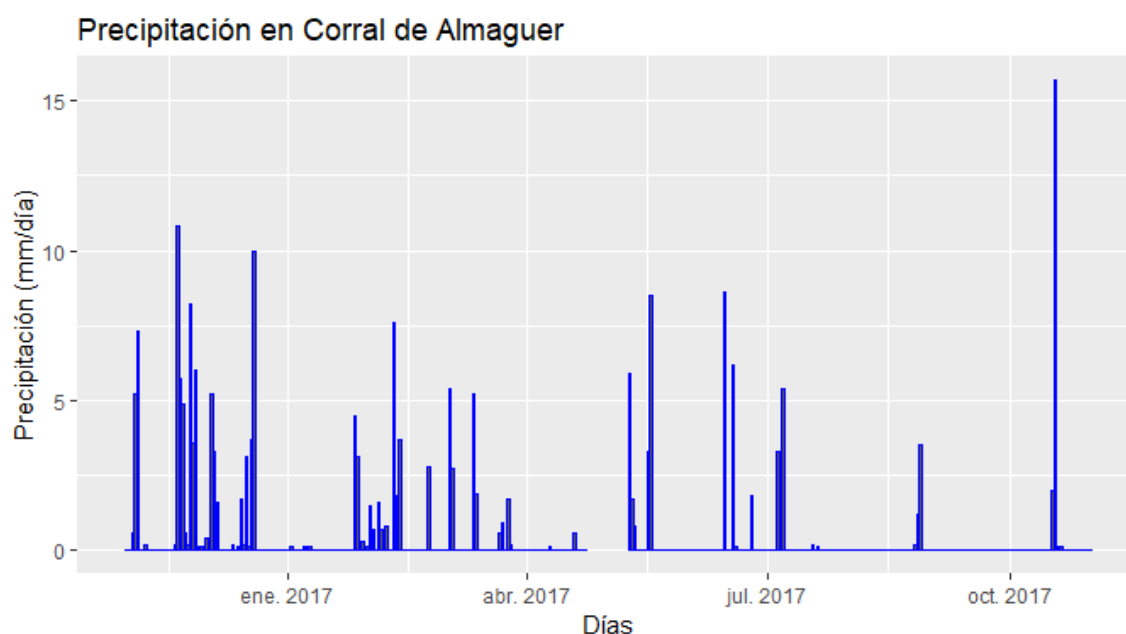


Figura A.2.6: Precipitación en la campaña 2016 – 2017

Después de calcular la precipitación, es necesario ver cómo la evapotranspiración afecta a la precipitación. ¿Cuál sería la lluvia útil? En el siguiente gráfico, se puede observar los mismos datos de precipitación junto con los datos obtenidos por regresión de la evapotranspiración (color rojo).

En este caso, para obtener la lluvia útil es necesario tener todos los datos de la evapotranspiración ya que se podrían ver huecos de información que importan en verano. Sin embargo, como hay valores que no se recogieron en el proceso (datos perdidos), se procede a usar un paquete desarrollado por Gary King, Matthew Blackwell (Harvard University) y James Honaker (The Pennsylvania State University) llamado Amelia II [111].

En resumen, Amelia II es un paquete de R destinado a la múltiple imputación de datos perdidos. Implementa una nueva expectativa maximización del algoritmo de Bootstrap que trabaja más rápido, con un número más grande de variables. Además, es más fácil de usar que las aproximaciones de la cadena de Markov el cual da las mismas respuestas.

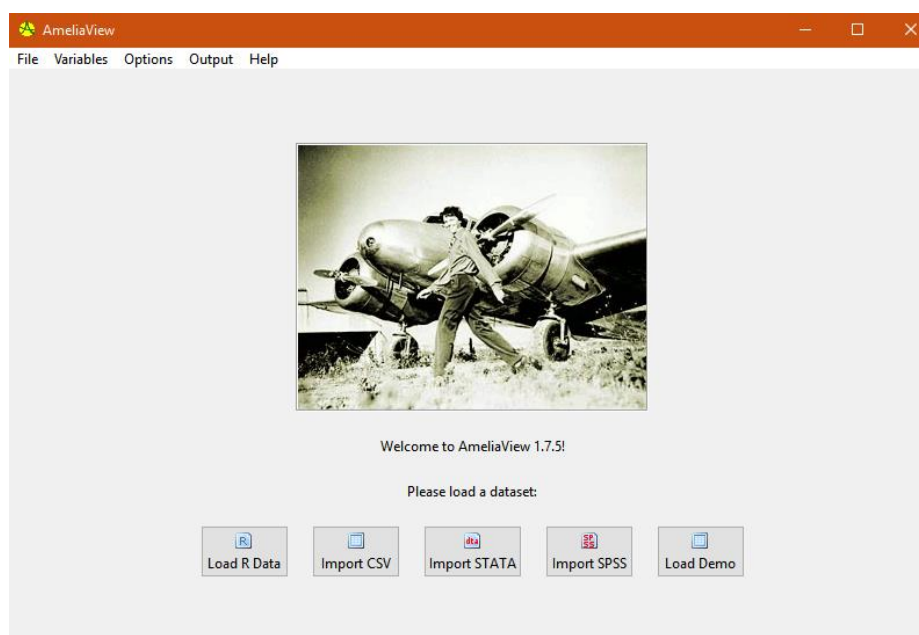


Figura A.2.7: Ejemplo del uso de Amelia II

El resultado de la aproximación de la evapotranspiración en la campaña 2016-2017 sería el siguiente:

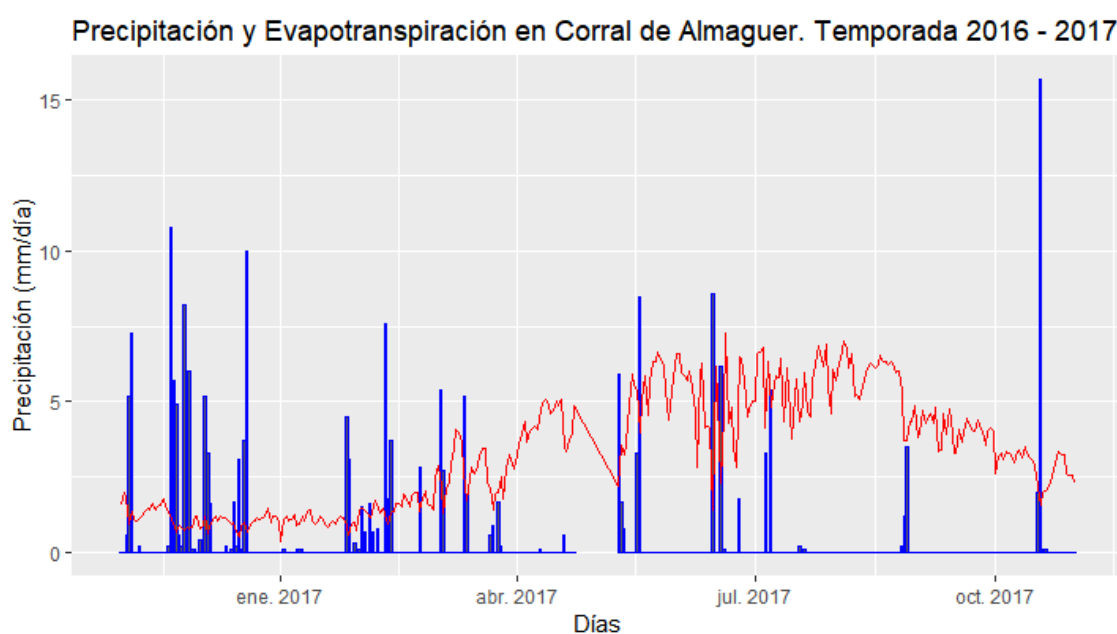


Figura A.2.8: Precipitación y evapotranspiración en la campaña 2016 – 2017

Los cálculos van a ser los siguientes: la lluvia útil (en verde) será 0 si la evapotranspiración es mayor a la precipitación. De lo contrario, la lluvia útil será la diferencia entre la precipitación y la evapotranspiración.

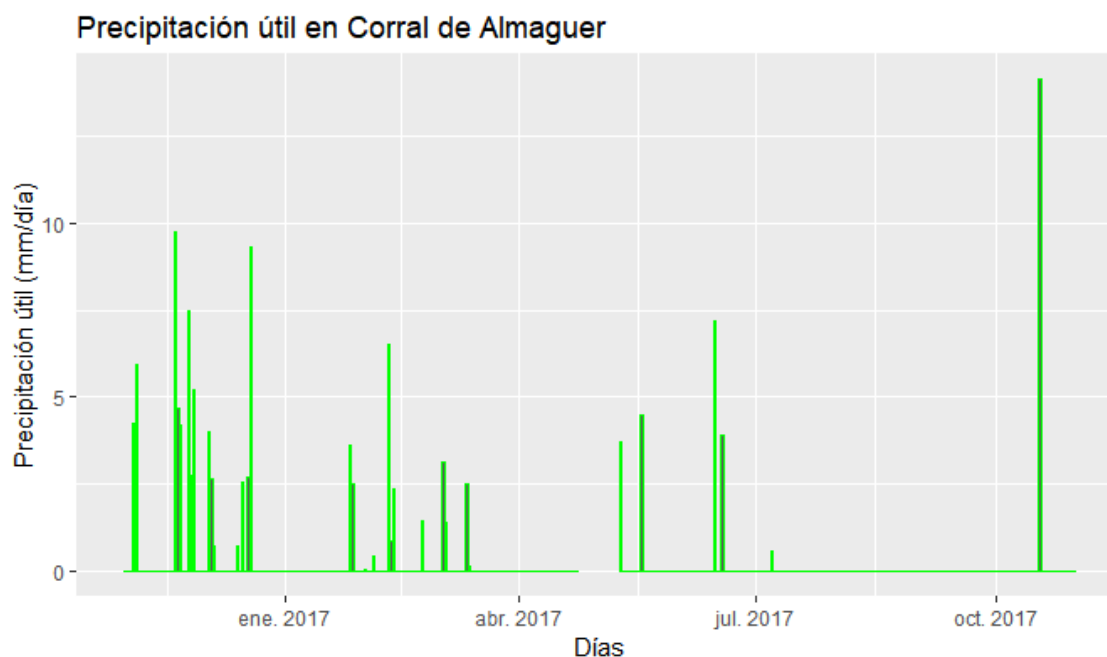


Figura A.2.9: Precipitación útil en la campaña 2016 – 2017

Aplicando las funciones “sum” en RStudio, se obtienen los datos totales de lluvia:

Precipitación total (azul)	201 L/m ²
Evapotranspiración de la lluvia total	74,83 L/m ²
Lluvia útil (verde)	126,17 L/m ²

Tabla A.2.3: Resultados precipitación, evapotranspiración y lluvia útil

Aunque los resultados parezcan nefastos (en verano no se pueden apreciar datos de lluvia útil), la gráfica no es exactamente real ya que hay otros factores que reducen la evapotranspiración (por ejemplo, cuando llueve, las nubes cubren parte de la radiación [112], las plantas absorben parte del agua que llueve...).

A continuación, para entender mejor la gráfica de la lluvia útil, se van a calcular los valores de agua por meses de acuerdo con los porcentajes de Lissarrague (en función de la lluvia útil que queda y del agua que se necesita para las plantas). Para ello, se utilizarán los conceptos de parada (8 mm de noviembre a finales de febrero), brotación y foliación (40 mm de marzo a mayo), fecundación y fructificación (172 mm en junio y mediados de julio) y envero y maduración (172 mm de mediados de julio hasta finales de octubre).

Los valores de la necesidad de agua se van a dividir de manera proporcional en las diferentes etapas salvo en el caso del envero y maduración ya que octubre tendrá la mitad de la parte correspondiente que se reparte en los meses de agosto y septiembre. Esto se debe a que justo nada más terminar la recolección, se produce la parada y la necesidad de agua vuelve a unos valores muy inferiores de nuevo.

La parte de octubre no se reparte en julio porque dicho mes tiene parte de fecundación y fructificación (86 litros por metro cuadrado) y otra parte de envero y maduración (43 litros por metro cuadrado). Es una cantidad que sobrepasa los demás meses por lo que es mejor que se reparta la cantidad de octubre en los otros.

Campaña 2016 - 2017												
Mes	Nov	Dec	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct
Lluvia útil	44.3	22.8	6.1	11.7	7.3	0	8.2	11.1	0.6	0	0	14.1
Necesidad	2	2	2	2	13.3	13.3	13.3	86	129	53.8	53.8	21.5
Necesidad diaria	0.07	0.06	0.06	0.07	0.44	0.45	0.44	2.86	4.16	1.73	1.79	0.69
Diferencia	+42.3	+20.8	+4.1	+9.7	-6	-13.3	-5.1	-74.9	-128.4	-53.8	-53.8	-7.4

Tabla A.2.4: Resultados finales en la campaña 2016 – 2017

Sin embargo, de acuerdo con los porcentajes de agua que establece Lissarrague, el agua no es suficiente para tener una buena producción ya que en los meses donde necesita más agua, la lluvia es menor o casi nula. También, es necesario enfatizar que, en los meses de parada, la lluvia útil sobrepasa a la necesidad ya que esta es muy pequeña en este periodo.

2.5. Una solución a corto plazo

Sin agua, la producción de las plantas no tiene futuro y no se podrá producir un vino de calidad, por lo tanto, la economía de Corral de Almaguer no saldrá adelante. Entonces, la solución que acogieron los agricultores con esperanza es el uso de los acuíferos o aguas subterráneas.

De acuerdo con Lucía de Stefano [113], las aguas subterráneas se han definido como invisibles, ocultas, silenciosas con un futuro impredecible ya que pasan desapercibidas.

Estas aguas representan el 98% de agua dulce que no está atrapada en los hielos de los polos y de los glaciares.

Un acuífero es el conjunto de rocas que permiten la permeabilidad del agua de lluvia infiltrándose desde la superficie y la pueden acumular en sus poros o grietas. A esta agua retenida en las estructuras rocosas se la conoce como agua subterránea. [114].

Las aguas subterráneas se han convertido en un elemento sustancial en el desarrollo socioeconómico de todo el mundo.

En España, gracias a la existencia de ríos y embalses, el agua subterránea supone solo un quinto del uso urbano, mientras que en otros países es el recurso principal. Por otro lado, en los años 70, se produjo una revolución en el uso de las aguas subterráneas en la que millones de agricultores españoles empezaron a construir pozos para su uso en la agricultura.

Hubo muchos avances científicos en el ámbito hidrológico y aumentaron las ventajas de las aguas subterráneas frente a las superficiales como la mayor disponibilidad del agua en lugares de interés (ya que puede que no haya un río o un lago cerca del cultivo, pero sí un acuífero), o la calidad del agua. Además, se empezaron a usar bombas sumergidas.

Sin embargo, como en muchos otros recursos, cuando se producen miles de perforaciones de pozos y se utilizan al mismo tiempo, se produce la degradación del recurso.

Anexo 3. Análisis del consumo de datos en la nube

El consumo de datos de una tarjeta SIM Hologram es gratuito siempre y cuando sea menos de 1 MB al mes. Esto es importante ya que es necesario inspeccionar cuándo y cómo se están gastando los datos contratados. Para visualizar el consumo, el administrador debe iniciar sesión en la nube de Hologram y clicar en uso (*Usage*). En esta sección existen dos apartados: Informes de uso y uso en directo (*Usage reports* y *live* respectivamente).

3.1. Informes de uso

En este apartado, se puede ver de manera general el consumo de datos en un gráfico en el que se puede modificar las fechas de inicio y fin del informe.

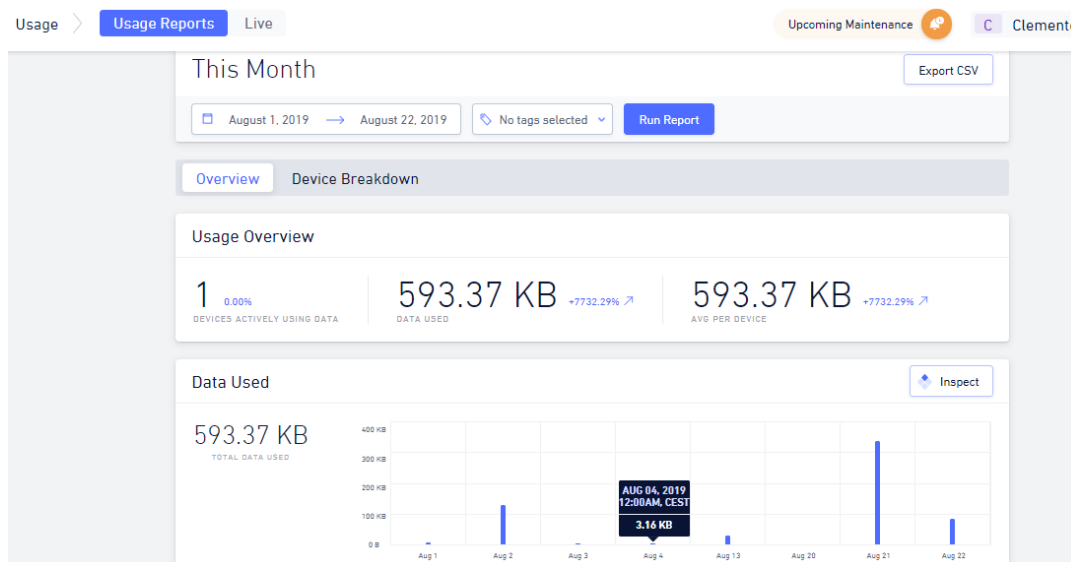


Figura A.3.1: Informes de uso

Además, para más detalle, se puede clicar en exportar CSV (*export CSV*) para descargarse el archivo CSV con toda la información en ese periodo de tiempo. Por otro lado, si se quiere ver en la web, se clicca en inspeccionar (*Inspect*). Se puede ver qué consumo hubo en cada día activo y en cuántas sesiones:

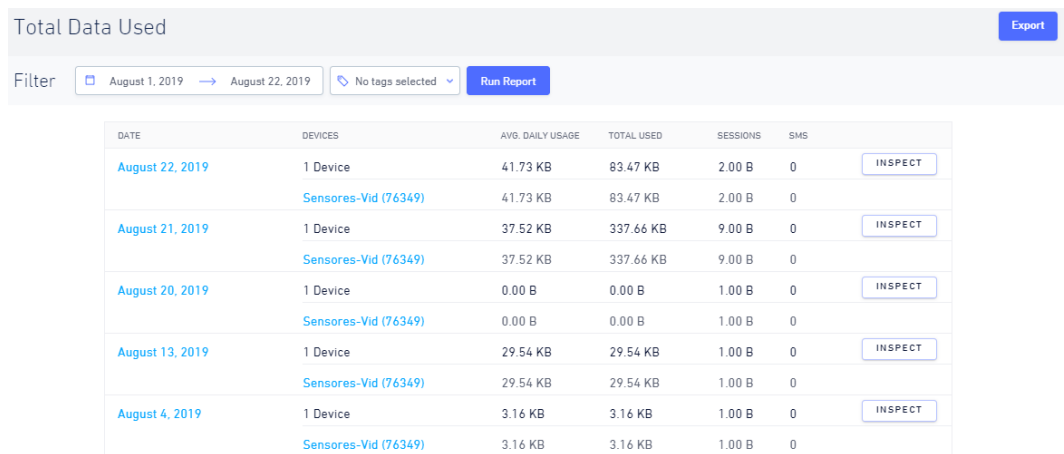


Figura A.3.2: Inspeccionar por días

3.2. Uso en directo

En este apartado, se puede ver en una gráfica el último mes de consumo y el usuario puede ampliar esa gráfica para visualizarlo en detalle tanto como desee. Además, debajo de la gráfica se encuentra el desglose de las sesiones.

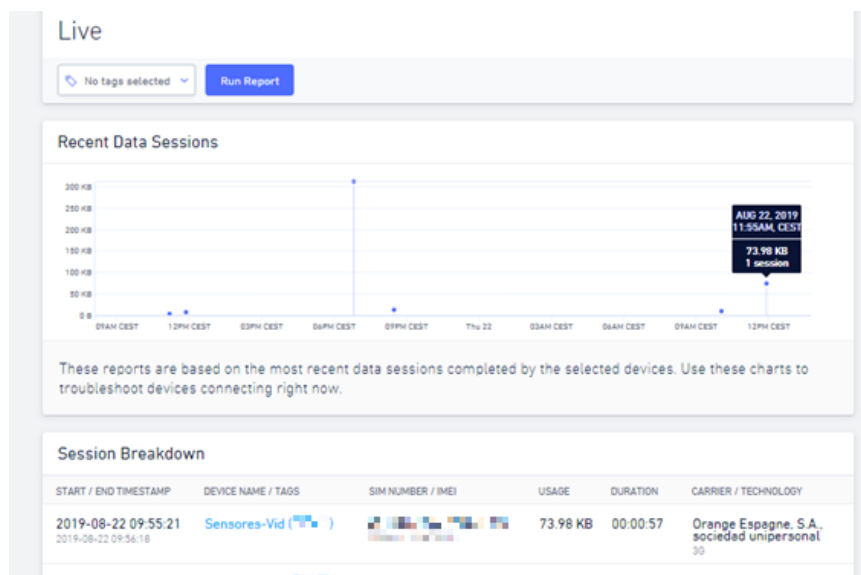


Figura A.3.3: Consumo en vivo

Anexo 4. Tabla de radiación solar extraterrestre [115]

HEMISFERIO NORTE													HEMISFERIO SUR												
Latitud	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic	
70	0.0	1.1	4.2	9.4	14.4	17.3	16.1	11.4	6.1	2.0	0.0	0.0	16.9	11.7	6.4	2.0	0.1	0.0	0.0	0.9	4.4	9.6	15.2	18.5	
68	0.0	1.5	4.8	9.8	14.4	17.1	15.9	11.7	6.6	2.4	0.3	0.0	16.7	12.0	6.9	2.4	0.3	0.0	0.0	1.3	4.9	10.0	15.3	18.2	
66	0.2	2.0	5.3	10.1	14.5	16.9	15.8	12.0	7.1	2.9	0.6	0.0	16.7	12.2	7.4	2.9	0.6	0.0	0.2	1.7	5.3	10.4	15.3	18.0	
64	0.6	2.4	5.8	10.5	14.7	16.8	15.8	12.2	7.5	3.5	1.0	0.2	16.7	12.6	7.9	3.4	1.0	0.2	0.5	2.2	5.9	10.7	15.5	17.9	
62	0.9	2.9	6.3	10.9	14.8	16.8	15.9	12.5	8.0	4.0	1.4	0.5	16.8	12.9	8.3	3.9	1.4	0.5	0.8	2.6	6.3	11.1	15.6	17.9	
60	1.3	3.4	6.8	11.2	14.9	16.8	16.0	12.8	8.4	4.4	1.8	0.9	16.9	13.2	8.8	4.4	1.8	0.8	1.2	3.1	6.8	11.5	15.8	17.9	
58	1.8	3.9	7.2	11.6	15.1	16.9	16.1	13.1	8.9	4.9	2.2	1.3	17.0	13.5	9.2	4.9	2.2	1.2	1.6	3.6	7.3	11.8	16.0	18.0	
56	2.2	4.4	7.7	11.9	15.3	16.9	16.2	13.3	9.3	5.4	2.7	1.7	17.1	13.8	9.6	5.4	2.7	1.6	2.0	4.0	7.8	12.2	16.1	18.0	
54	2.7	4.9	8.2	12.2	15.4	16.9	16.2	13.6	9.7	5.9	3.2	2.1	17.2	14.0	10.0	5.9	3.1	2.0	2.4	4.5	8.2	12.5	16.3	18.1	
52	3.1	5.4	8.6	12.6	15.6	17.0	16.4	13.8	10.1	6.4	3.7	2.6	17.3	14.3	10.4	6.4	3.6	2.4	2.9	5.0	8.7	12.8	16.4	18.1	
50	3.6	5.9	9.1	12.9	15.7	17.0	16.4	14.0	10.5	6.9	4.2	3.1	17.4	14.5	10.9	6.8	4.1	2.9	3.3	5.5	9.1	13.1	16.6	18.2	
48	4.1	6.4	9.5	13.1	13.8	17.1	16.5	14.2	10.9	7.4	4.7	3.6	17.5	14.8	11.2	7.3	4.5	3.3	3.8	6.0	9.5	13.4	16.7	18.2	
46	46.1	6.9	9.9	13.4	16.0	17.1	16.6	14.4	11.2	7.8	5.1	4.0	17.6	15.0	11.6	7.8	5.0	3.8	4.2	6.4	9.9	13.7	16.8	18.2	
44	5.1	7.3	10.3	13.7	16.0	17.1	16.6	14.7	11.6	8.3	5.7	4.5	17.6	15.2	12.0	8.2	5.5	4.3	4.7	6.9	10.3	13.9	16.9	18.2	
42	5.6	7.8	10.7	13.9	16.1	17.1	16.7	14.8	11.9	8.7	6.2	5.1	17.7	15.4	12.3	8.7	6.0	4.7	5.2	7.3	10.7	14.2	17.0	18.2	
40	6.1	8.3	11.1	14.2	16.2	17.1	16.7	15.0	12.2	9.2	6.7	5.6	17.7	15.6	12.6	9.1	6.4	5.2	5.7	7.8	11.1	14.4	17.1	18.2	
38	6.6	8.8	11.5	14.4	16.3	17.1	16.7	15.1	12.5	9.6	7.1	6.0	17.7	15.7	12.9	9.5	6.9	5.7	6.2	8.2	11.4	14.6	17.1	18.2	
36	7.1	9.2	11.8	14.6	16.3	17.0	16.7	15.3	12.9	10.0	7.6	6.6	17.7	15.9	13.2	9.9	7.4	6.2	6.6	8.7	11.8	14.8	17.1	18.1	
34	7.6	9.7	12.2	14.7	16.3	17.0	16.7	15.3	13.1	10.4	8.1	7.1	17.7	16.0	13.5	10.3	7.8	6.6	7.1	9.1	12.1	15.0	17.1	18.1	
32	8.1	10.1	12.5	14.5	16.3	16.9	16.6	15.5	13.4	10.9	8.6	7.6	17.7	16.1	13.8	10.7	8.3	7.1	7.6	9.5	12.4	15.1	17.1	18.0	
30	8.6	10.5	12.8	15.0	16.3	16.8	16.6	15.5	13.6	11.3	9.1	8.1	17.6	16.2	14.0	11.1	8.7	7.6	8.0	9.9	12.7	15.3	17.1	17.9	
28	9.1	10.9	13.1	15.1	16.3	16.7	16.5	15.6	13.8	11.6	9.5	8.6	17.6	16.2	14.2	11.5	9.2	8.0	8.4	10.3	13.0	15.4	17.1	17.8	
26	9.6	11.3	13.4	15.3	16.3	16.6	16.4	15.6	14.1	12.0	10.0	9.1	17.5	16.3	14.4	11.8	9.6	8.5	8.9	10.7	13.3	15.5	17.1	17.7	
24	10.0	11.8	13.7	15.3	16.2	16.4	16.3	15.6	14.2	12.3	10.4	9.5	17.3	16.3	14.6	12.2	10.0	8.9	9.3	11.1	13.5	15.6	17.0	17.6	
22	10.5	12.1	13.9	15.4	16.1	16.3	16.2	15.7	14.4	12.7	10.9	10.0	17.2	16.4	14.8	12.5	10.4	9.4	9.8	11.5	13.8	15.7	16.9	17.4	
20	10.9	12.5	14.2	15.5	16.0	16.1	16.0	15.6	14.6	13.0	11.3	10.4	17.1	16.3	14.9	12.8	10.9	9.8	10.2	11.8	14.0	15.8	16.8	17.2	
18	11.4	12.9	14.4	15.5	15.9	16.0	15.9	15.6	14.7	13.3	11.7	10.9	16.9	16.3	15.1	13.1	11.2	10.2	10.6	12.2	14.2	15.8	16.7	17.0	
16	11.8	13.2	14.6	15.6	15.8	15.8	15.7	15.6	14.9	13.6	12.1	11.4	16.8	16.3	15.2	13.4	11.6	10.7	11.0	12.5	14.4	15.8	16.6	16.8	
14	12.2	13.5	14.7	15.6	15.7	15.6	15.6	15.5	15.0	13.8	12.5	11.8	16.6	16.2	15.3	13.6	12.0	11.1	11.4	12.8	14.5	15.8	16.4	16.6	
12	12.6	13.8	14.9	15.5	15.5	15.3	15.3	15.4	15.1	14.1	12.9	12.2	16.4	16.2	15.4	13.9	12.3	11.5	11.8	13.1	14.7	15.8	16.2	16.3	
10	13.0	14.1	15.1	15.5	15.3	15.1	15.1	15.3	15.1	14.3	13.2	12.7	16.1	16.0	15.4	14.1	12.7	11.9	12.2	13.4	14.8	15.7	16.0	16.1	
8	13.4	14.4	15.2	15.4	15.1	14.8	14.9	15.2	15.2	14.5	13.6	13.1	15.9	15.9	15.5	14.3	13.0	12.2	12.5	13.6	14.9	15.7	15.8	15.8	
6	13.8	14.6	15.3	15.3	14.9	14.6	14.7	15.1	15.2	14.7	13.9	13.4	15.6	15.8	15.5	14.5	13.3	12.6	12.9	13.9	15.0	15.6	15.6	15.5	
4	14.1	14.9	15.3	15.3	14.7	14.3	14.4	14.9	15.2	14.9	14.2	13.8	15.3	15.6	15.5	14.7	13.6	13.0	13.2	14.1	15.1	15.5	15.3	15.2	
2	14.4	15.1	15.4	15.1	14.4	14.0	14.1	14.7	15.2	15.1	14.5	14.2	15.1	15.5	15.5	14.9	13.9	13.3	13.5	14.4	15.1	15.4	15.1	14.9	
0	14.8	15.3	15.5	15.0	14.2	13.6	13.8	14.6	15.2	15.3	14.8	14.5	14.8	15.3	15.5	15.0	14.2	13.6	13.8	14.6	15.2	15.3	14.8	14.5	

Anexo 5. Glosario de acrónimos y definiciones

❖ Acrónimos

- ADC: Conversor Analógica - Digital
- AI: Inteligencia Artificial
- API: Interfaz de programación de aplicaciones
- CSV: Valores separados por comas
- GNU: *GNU's not Unix*. El diseño de GNU es de tipo *Unix* pero difiere de éste ya que es software libre y no contiene código *Unix*.
- GPS: Sistema de Posicionamiento Global
- GSM: Sistema global para las comunicaciones móviles
- I2C: Circuito inter-integrado
- IDC: Corporación de Datos Internacionales
- IDE: Entorno de Desarrollo Integrado
- IIoT: Internet Industrial de las Cosas
- IoT: Internet de las Cosas
- IP: Protocolo de Internet
- *Modem*: Modulador – Demodulador
- OS: Operating System
- PIR: Sensor Infrarrojo Pasivo
- RFID: Identificación por radiofrecuencia
- SSH: Secure SHell
- SSL: Secure Socket Layer

- Telnet: Red de telecomunicación
- TIC: Tecnologías de la Información y la Comunicación
- TCP: Protocolo de Control de Transporte
- UDP: Protocolo de Datagramas de Usuario
- VNC: Computación de red virtual
- WiFi: Fidelidad inalámbrica
- WSN: Redes de Sensores Inalámbricas

❖ Definiciones

- **Módem:** es un dispositivo que convierte las señales digitales en analógicas (modulación) y viceversa (desmodulación) que sirve para enviar la señal moduladora mediante otra señal llamada portadora. [101]
- **Big Data:** En español, datos masivos, son archivos de información de gran volumen, alta velocidad y/o gran variedad que exigen formas rentables e innovadoras de procesamiento de información que permitan una mejor comprensión, toma de decisiones y automatización de procesos [102].
- **Unix:** es un sistema operativo portable, multitarea y multiusuario; desarrollado en 1969 por un grupo de empleados de los laboratorios Bell de AT&T, entre los que figuran Dennis Ritchie, Ken Thompson y Douglas McIlroy [103].