

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**DISEÑO DE RED LORAWAN EN CULTIVO DE FRESAS PARA
MONITOREO DE HUMEDAD DEL SUELO.**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN
SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

ERIKA PAMELA SILVA GÓMEZ

erika.silva@epn.edu.ec

DIRECTOR: Sang Guun Yoo, Ph.D.

sang.yoo@epn.edu.ec

Quito, febrero 2021

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Erika Pamela Silva Gómez, bajo mi supervisión.



Sang Guun Yoo, Ph.D.
DIRECTOR DE PROYECTO

DECLARACIÓN

Yo Erika Pamela Silva Gómez, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.



Erika Pamela Silva Gómez

DEDICATORIA

A mi padres por ser el pilar fundamental en mi vida. Que este trabajo de titulación sea un resultado más de todo el esfuerzo y dedicación que han puesto sobre mí. En especial a ti mamá, sin tu ayuda, nada de lo que he logrado hubiera sido posible. Me has guiado por el buen camino con principios y valores, todo te lo debo a ti.

A mi hija, por ser mi motor, mi fuerza y mis ganas de seguir adelante. Eres la motivación más grande en mi vida.

Erika Silva

AGRADECIMIENTOS

Agradezco a mi Dios por darme la vida, por su infinito amor y bondad, por forjarme en el camino del bien con principios y valores. Gracias por todas las bendiciones que me ha concedido.

A mis padres, por su amor sin límites, por su apoyo en cada paso que doy, por enseñarme a decir "sí puedo". Gracias a ustedes soy lo que soy, gracias por todo su esfuerzo y trabajo.

A mi hermano, por ser mi ejemplo, mi gran maestro. Gracias por tus enseñanzas y apoyo incondicional.

A mi novio, por su amor y apoyo en los momentos más difíciles, por ser mi amigo, mi confidente, mi compañero de vida. Gracias sobre todo por tu paciencia.

A mi familia, amigos y compañeros, por su compañía, amistad y apoyo en los buenos y malos momentos.

A mis maestros, por brindarme sus conocimientos y más que nada por los grandes consejos en lo académico, profesional y personal.

A la Escuela Politécnica Nacional, por la oportunidad de permitirme formar una carrera profesional.

A mi director de proyecto Ing. Sang Guun Yoo, Ph.D, por sus conocimientos, tiempo y confianza puesta en mí ayudándome a culminar con éxito este proyecto.

A Hanna, mi hija por llegar a cambiarme la vida, por llegar en el momento indicado, por iluminar mi vida con tu sonrisa. Muchos pensaron que ibas a ser un freno en mi vida, pero hoy eres el motor que me mantiene de pie. Gracias por darme las fuerzas para seguir adelante y cumplir mis sueños.

Erika Silva

ÍNDICE DE CONTENIDO

CERTIFICACIÓN	2
DECLARACIÓN.....	3
DEDICATORIA	4
AGRADECIMIENTOS	5
ÍNDICE DE TABLAS.....	9
ÍNDICE DE FIGURAS.....	11
RESUMEN	15
ABSTRACT	16
CAPÍTULO 1. INTRODUCCIÓN	17
1.1 PLANTEAMIENTO DEL PROBLEMA	17
1.2 JUSTIFICACIÓN PRÁCTICA	18
1.3 JUSTIFICACIÓN TEÓRICA	18
1.4 OBJETIVOS.....	18
1.4.1 <i>Objetivo General</i>	18
1.4.2 <i>Objetivos Específicos</i>	19
CAPÍTULO 2. MARCO TEÓRICO	19
2.1 INTERNET DE LAS COSAS (IoT)	19
2.1.1 <i>Arquitectura de Sistemas de IoT</i>	20
2.1.2 <i>Áreas de aplicación</i>	20
2.2 AGRICULTURA DE PRECISIÓN	21
2.3 TECNOLOGÍA LORA.....	21
2.4 PROTOCOLO LORAWAN	22
2.4.1 <i>Arquitectura LoRaWAN</i>	22
2.4.2 <i>Cíases LoRaWAN</i>	23
2.4.3 <i>Activación de los dispositivos finales</i>	24
2.4.4 <i>OTAA (Over-The-Air Activation)</i>	25
2.4.5 <i>ABP (Activation By Personalization)</i>	27
CAPÍTULO 3. METODOLOGÍA Y TECNOLOGÍAS.....	28
3.1 METODOLOGÍA.....	28
3.2 TECNOLOGÍAS UTILIZADAS.....	29

3.2.1	<i>Protocolos</i>	29
3.2.2	<i>Diagramación y Prototipado</i>	30
3.2.3	<i>Lenguajes de Programación</i>	31
3.2.4	<i>Firebase</i>	32
3.2.4.1	Base de datos.....	32
3.2.4.2	Autenticación.....	34
3.2.4.3	Despliegue.....	34
3.2.5	<i>Frameworks y Bibliotecas</i>	35
3.2.6	<i>Control de Versiones</i>	36
3.2.7	<i>Entornos de Desarrollo</i>	37
CAPÍTULO 4. DESARROLLO DE LA SOLUCIÓN		38
4.1	DESCRIPCIÓN DEL CASO DE ESTUDIO.....	38
4.1.1	<i>Descripción del cultivo de fresas</i>	38
4.1.2	<i>Proceso de cuidado del cultivo de fresas</i>	39
4.1.3	<i>Descripción del problema</i>	41
4.2	DISEÑO DE LA RED LORAWAN.....	42
4.2.1	<i>Diseño de la red LoRaWAN para el caso de estudio</i>	42
4.2.2	<i>Equipamiento para la implementación de la red LoRaWAN</i>	44
4.2.2.1	LoRaWAN Gateway.....	45
4.2.2.2	Sensor de humedad del suelo.....	46
4.3	IMPLEMENTACIÓN DE LA RED LORAWAN.....	48
4.3.1	<i>Configuración e Instalación de equipos de la red LoRaWAN</i>	48
4.3.1.1	Configuración del gateway LPS8.....	48
4.3.1.2	Configuración e instalación de los sensores LSE01.....	55
4.3.2	<i>Comprobación del funcionamiento de los sensores instalados</i>	61
4.4	DESARROLLO DEL APLICATIVO WEB.....	62
4.4.1	<i>Arquitectura de la aplicación</i>	62
4.4.2	<i>Product Backlog</i>	63
4.4.3	<i>Sprint 1</i>	66
4.4.3.1	Sprint Planning.....	66
4.4.3.2	Codificación.....	67
4.4.3.3	Sprint Review.....	77
4.4.3.4	Despliegue.....	78
4.4.4	<i>Sprint 2</i>	78
4.4.4.1	Sprint Planning.....	78
4.4.4.2	Codificación.....	80
4.4.4.3	Sprint Review.....	88
4.4.4.4	Despliegue.....	89

4.4.5	<i>Sprint 3</i>	90
4.4.5.1	Sprint Planning.....	90
4.4.5.2	Codificación.....	92
4.4.5.3	Sprint Review.....	98
4.4.5.4	Despliegue.....	99
4.4.6	<i>Sprint 4</i>	100
4.4.6.1	Sprint Planning.....	100
4.4.6.2	Codificación.....	102
4.4.6.3	Sprint Review.....	110
4.4.6.4	Despliegue.....	111
CAPÍTULO 5. RESULTADOS Y DISCUSIÓN		113
5.1	PRUEBAS DE FUNCIONALIDAD.....	113
5.1.1	<i>Casos de prueba</i>	113
5.1.2	<i>Resultados Globales</i>	119
5.2	PRUEBAS DE USABILIDAD.....	121
5.2.1	<i>Resultados individuales</i>	122
5.2.2	<i>Resultados Globales</i>	126
5.3	PRUEBA DE COBERTURA.....	127
CAPÍTULO 6. CONCLUSIONES Y RECOMENDACIONES		133
6.1	CONCLUSIONES.....	133
6.2	RECOMENDACIONES.....	134
BIBLIOGRAFÍA		136
ANEXOS		140
ANEXO 1: MOCKUPS DE LA APLICACIÓN WEB.....		140
ANEXO 2: RESULTADOS DE LOS CASOS DE PRUEBA		140
ANEXO 3: ENCUESTA DE PRUEBAS DE FUNCIONALIDAD.....		140
ANEXO 4: ENCUESTA DE PRUEBAS DE USABILIDAD.....		140
ANEXO 5: RESULTADOS DE LAS ENCUESTAS DE PRUEBAS DE FUNCIONALIDAD		140
ANEXO 6: RESULTADOS DE LAS ENCUESTAS DE PRUEBAS DE USABILIDAD		141
ANEXO 7: VIDEO DEMOSTRATIVO DE LA FUNCIONALIDAD DE LA APLICACIÓN WEB		141

ÍNDICE DE TABLAS

Tabla 1. Arquitectura IoT de 4 capas.	20
Tabla 2. Principales características de la Tecnología LoRa,	22
Tabla 3. Principales Características de las Clases LoRaWAN.	24
Tabla 4. Datos almacenados en el dispositivo final.	25
Tabla 5. Campos del mensaje Join-Request. Fuente [15].	26
Tabla 6. Campos del mensaje Join-Accept. Fuente [15].	27
Tabla 7. Pasos de conexión con el método OTAA.	27
Tabla 8. Pasos de conexión con el método ABP.	28
Tabla 9. Protocolo utilizado.	30
Tabla 10. Herramientas de diagramación y prototipado utilizadas.	31
Tabla 11. Lenguajes de programación utilizados.	32
Tabla 12. Base de datos utilizada.	33
Tabla 13. Herramienta de autenticación utilizada.	34
Tabla 14. Herramienta de despliegue utilizada.	35
Tabla 15. Frameworks para front-end utilizados.	35
Tabla 16. Frameworks y bibliotecas para back-end utilizados.	36
Tabla 17. Biblioteca para comunicación entre back-end y front-end utilizada.	36
Tabla 18. Control de versiones utilizado.	37
Tabla 19. Entorno de desarrollo utilizado.	37
Tabla 20. Características del caso de estudio.	39
Tabla 21. Especificaciones técnicas del Gateway LPS8.	46
Tabla 22. Características y aplicaciones del Gateway LPS8.	46
Tabla 23. Especificaciones técnicas del Sensor LES01.	47
Tabla 24. Características y aplicaciones del sensor LES01.	48
Tabla 25. Indicadores LED del dispositivo LPS8. Fuente [48].	49
Tabla 26. Información de la red creada por el dispositivo LPS8.	50
Tabla 27. Credenciales del dispositivo LPS8.	51
Tabla 28. Claves del dispositivo LSE01-01.	57
Tabla 29. Componentes del uplink payload.	58
Tabla 30. Claves del dispositivo LSE01-02.	60
Tabla 31. Product Backlog de la Aplicación Web.	65
Tabla 32. Sprint Planning - Requerimientos para el Sprint 1.	66
Tabla 33. Tareas del Sprint 1.	67
Tabla 34. Sprint Review del Sprint 1.	78

Tabla 35. Sprint Planning - Requerimientos para el Sprint 2.....	79
Tabla 36. Tareas del Sprint 2.....	80
Tabla 37. Niveles de humedad de suelo.....	87
Tabla 38. Sprint Review del Sprint 2.....	89
Tabla 39. Sprint Planning - Requerimientos para el Sprint 3.....	90
Tabla 40. Tareas del Sprint 3.....	92
Tabla 41. Sprint Review del Sprint 3.....	99
Tabla 42. Sprint Planning - Requerimientos para el Sprint 4.....	100
Tabla 43. Tareas del Sprint 4.....	101
Tabla 44. Sprint Review del Sprint 4.....	110
Tabla 45. Formato de caso de prueba para pruebas de funcionalidad.....	113
Tabla 46. Caso de Prueba N°1 – Creación de una cuenta e inicio de sesión.....	114
Tabla 47. Caso de Prueba N°2 – Registro de cultivo.....	115
Tabla 48. Caso de Prueba N°3 – Registro de riego.....	116
Tabla 49. Caso de Prueba N°4 – Búsqueda de registros de los valores de los sensores por fecha.....	116
Tabla 50. Caso de Prueba N°5 – Editar un registro de riego.....	117
Tabla 51. Caso de Prueba N°6 – Eliminar un registro de valores de los sensores.....	118
Tabla 52. Caso de Prueba N°7 – Visualización de valores de los sensores del cultivo.....	119
Tabla 53. Caso de Prueba N°8 – Visualización de mensajes de advertencia cuando la humedad está fuera del rango establecido.....	119
Tabla 54. Puntajes obtenidos según la escala SUS por cada usuario.....	127
Tabla 55. Puntos de ubicación de los dispositivos de la red LoRaWAN.....	128

ÍNDICE DE FIGURAS

Figura 1. Arquitectura LoRaWAN. Fuente [16].	23
Figura 2. Arquitectura en capas LoRaWAN. Fuente [17].	24
Figura 3. Método de Activación OTAA. Fuente [17].	26
Figura 4. Método de Activación ABP. Fuente [17].	27
Figura 5. Mapa de acceso al Cultivo de Fresas Renan Silva.	38
Figura 6. Plano de referencia del cultivo de fresas tomado de Google Earth. Fuente [38]	39
Figura 7. Diseño de la red LoRaWAN.	44
Figura 8. LPS8 Indoor LoRaWAN Gateway. Fuente [48].	45
Figura 9. Sensor de humedad del suelo LSE01 LoRaWAN. Fuente [49].	47
Figura 10. Sensor LSE01 en una red LoRaWAN. Fuente [49].	48
Figura 11. Dispositivo LPS8 encendido.	49
Figura 12. Diagrama de conexión del dispositivo LPS8.	50
Figura 13. Red Wifi creada por el dispositivo LPS8.	51
Figura 14. Interfaz de inicio de sesión para la configuración del dispositivo LPS8.	52
Figura 15. Portal de configuración del dispositivo LPS8.	52
Figura 16. Interfaz de configuración de la red LoRaWAN.	53
Figura 17. Página principal del servidor TTN.	53
Figura 18. Registro del gateway en el servidor TTN.	54
Figura 19. Resumen del gateway creado en el servidor TTN.	54
Figura 20. Portal de configuración del dispositivo LPS8 después de la configuración.	54
Figura 21. Dispositivo LPS8 después de configurar la red LoRaWAN.	55
Figura 22. Creación de una aplicación en la consola del servidor TTN.	56
Figura 23. App EUIS de los sensores LSE01-01 y LSE01-02.	56
Figura 24. Registro del dispositivo LSE01-01.	57
Figura 25. Estado del dispositivo LSE01-01.	58
Figura 26. Formato de payload.	59
Figura 27. Mensajes intercambiados entre el sensor LSE01-01 y el servidor en el proceso de unión a la red.	59
Figura 28. Registro del dispositivo LSE01-02.	60
Figura 29. Estado del dispositivo LSE01-02.	61
Figura 30. Mensajes intercambiados entre el sensor LSE01-02 y el servidor en el proceso de unión a la red.	61
Figura 31. Mensajes recibidos desde el sensor LSE01-01.	62
Figura 32. Mensajes recibidos desde el sensor LSE01-02.	62

Figura 33. Arquitectura de la aplicación web.....	63
Figura 34. Estructura NoSQL de los datos en Cloud Firestore.....	68
Figura 35. Proyectos creados en Firebase.....	69
Figura 36. Asignación de nombre al nuevo proyecto.....	69
Figura 37. Consola de Firebase para el proyecto Fresas Santa Rosa.....	70
Figura 38. Plataformas disponibles.....	70
Figura 39. Pasos para agregar una aplicación web en Firebase.....	71
Figura 40. Paso 1: Registro de la aplicación web en Firebase.....	71
Figura 41. Paso 2: Adición de SDK de Firebase.....	72
Figura 42. SDK añadidos al proyecto.....	72
Figura 43. Contenido del archivo <i>init.js</i>	72
Figura 44. Paso 3: Instalación de CLI de Firebase.....	73
Figura 45. Instalación de Firebase Tools.....	73
Figura 46. Requisito R01 – Pantalla de creación de cuentas.....	74
Figura 47. Requisito R01 – Validación de campos.....	74
Figura 48. Requisito R02 – Pantalla de inicio de sesión.....	75
Figura 49. Requisito R02 – Inicio de sesión con credenciales incorrectas.....	76
Figura 50. Requisito R02 – Inicio de sesión con credenciales correctas.....	77
Figura 51. Pantalla principal de la aplicación.....	77
Figura 52. Registro de usuarios en la base de datos.....	77
Figura 53. Interfaz del módulo de cultivos sin registros.....	81
Figura 54. Requisito R03 – Pantalla de Registro de Cultivos.....	81
Figura 55. Requisito R03 – Validación de campos.....	82
Figura 56. Registro de cultivo en la base de datos de Cloud Firestore.....	82
Figura 57. Requisito R04 – Visualización de información de cultivos.....	83
Figura 58. Requisito R05 – Edición de la información de cultivos.....	83
Figura 59. Interfaz de la pantalla principal sin sensores.....	84
Figura 60. ID del dispositivo y clave de acceso de la aplicación registrados en TTN.....	84
Figura 61. Datos del sensor recibidos en la consola de la aplicación.....	85
Figura 62. Requisito R06 - Visualización de los valores de los sensores.....	85
Figura 63. Mensaje de creación de registro de los sensores en la base de datos.....	86
Figura 64. Información de los sensores registrados en Cloud Firestore.....	86
Figura 65. Contenido Volumétrico de Agua.....	87
Figura 66. Mensaje de error para valores de humedad bajos.....	87
Figura 67. Requisito R07 – Notificación de riego en el cultivo.....	88

Figura 68. R03 – Registro de información de cultivos. Mensaje de confirmación de registro de datos.....	92
Figura 69. R07 – Notificación de riego en el cultivo – Mensaje de información para valores altos de humedad.	93
Figura 70. Requisito R08 – Pantalla de visualización de los sensores conectados en la aplicación.....	94
Figura 71. Requisito R09 – Pantalla de búsqueda de registros de datos de sensores.	94
Figura 72. Ejemplo de búsqueda de registros de los sensores.	95
Figura 73. Botón regresar de la pantalla de búsqueda de los datos de sensores.....	95
Figura 74. Botón eliminar de la pantalla de búsqueda de los datos de sensores.	96
Figura 75. Interfaz del módulo de Riego.	96
Figura 76. Pantalla para agregar nuevo registro de riego.	97
Figura 77. Requisito R11 – Registro de información de riegos.....	98
Figura 78. Registro de cultivo en la base de datos de Cloud Firestore.....	98
Figura 79. R10 – Eliminación de registros de datos de los sensores. Mensaje de confirmación de eliminación de registros.	102
Figura 80. Requisito R12 – Visualización de información de riegos.	103
Figura 81. Tabla de registros de riego con el botón editar.	103
Figura 82. Requisito R13 – Edición de información de riegos.	104
Figura 83. Botón eliminar de la pantalla de búsqueda de los registro de riego.....	104
Figura 84. Mensaje de confirmación antes de eliminar registros de riego.	105
Figura 85. Requisito R09 – Pantalla de búsqueda de registros de datos de sensores. ..	105
Figura 86. Ejemplo de búsqueda de registros de los sensores.	106
Figura 87. Requisito R16 – Restablecimiento de contraseña.....	107
Figura 88. Email de restablecimiento de contraseña.....	107
Figura 89. Interfaz de Cambio de Contraseña de Firebase Authentication.....	108
Figura 90. Plantilla del correo electrónico para el cambio de contraseña de Firebase Authentication.....	108
Figura 91. Requisito R17 – Visualización de la pantalla de inicio de la aplicación web. .	109
Figura 92. Ejemplo de Visualización de la pantalla de inicio de la aplicación web.....	109
Figura 93. Contenido del archivo <i>package.json</i> para el despliegue de la aplicación.	111
Figura 94. Contenido del archivo <i>Procfile</i> para el despliegue de la aplicación.....	111
Figura 95. Aplicación web desplegada en Heroku.	112
Figura 96. Distribución de edades en las pruebas de la aplicación.	120
Figura 97. Distribución del área de trabajo en las pruebas de la aplicación.	120
Figura 98. Gráfico de resultados de la encuesta de pruebas de funcionalidad.....	121

Figura 99. Resultados globales de la encuesta de funcionalidad.	121
Figura 100. Resultados de la Pregunta N°1 de la Encuesta de Usabilidad.	122
Figura 101. Resultados de la Pregunta N°2 de la Encuesta de Usabilidad.	123
Figura 102. Resultados de la Pregunta N°3 de la Encuesta de Usabilidad.	123
Figura 103. Resultados de la Pregunta N°4 de la Encuesta de Usabilidad.	123
Figura 104. Resultados de la Pregunta N°5 de la Encuesta de Usabilidad.	124
Figura 105. Resultados de la Pregunta N°6 de la Encuesta de Usabilidad.	124
Figura 106. Resultados de la Pregunta N°7 de la Encuesta de Usabilidad.	124
Figura 107. Resultados de la Pregunta N°8 de la Encuesta de Usabilidad.	125
Figura 108. Resultados de la Pregunta N°9 de la Encuesta de Usabilidad.	125
Figura 109. Resultados de la Pregunta N°10 de la Encuesta de Usabilidad.....	125
Figura 110. Gráfico de los puntajes promedio para cada pregunta de la encuesta de usabilidad según la escala SUS.....	126
Figura 111. Distancia medida entre el gateway y el cultivo de fresas. Tomada desde Google Maps.....	128
Figura 112. Sensor instalado en el suelo de manera horizontal.	129
Figura 113. Sensor LSE01-01 instalado en el cultivo de fresas.	130
Figura 114. Sensor LSE01-02 instalado en el cultivo de fresas.	131
Figura 115. Recepción de datos de los sensores instalados.....	132

RESUMEN

El Internet de las Cosas o *Internet of Things* (IoT, por sus siglas en inglés) aplicado a la agricultura ha permitido lograr un desarrollo tecnológico en este sector a través del uso de tecnologías como las redes LPWAN. Esta tecnología permite obtener un control más exacto de la información en tiempo real logrando una mejor toma de decisiones en la gestión de los cultivos con el fin de mejorar la calidad de estos.

En el presente trabajo se tiene como objetivo, la aplicación de la tecnología LoRa en un cultivo de fresas para realizar el monitoreo de la humedad del suelo a través de una aplicación web. La aplicación permite la visualización de los datos obtenidos a partir de los sensores instalados en el suelo. Dependiendo de los niveles de humedad del suelo, muestra mensajes hacia el usuario que le permiten conocer cuando el cultivo necesita agua.

Para el desarrollo de la aplicación web se utilizó la metodología SCRUM. Esto permitió tener la participación de los interesados, obteniendo una retroalimentación constante. Para la codificación se utilizó los frameworks de Node JS, Express JS y Semantic UI para el desarrollo de front-end y back-end en conjunto con la plataforma de Firebase como gestor de autenticación y base de datos.

Al culminar el proyecto, se obtuvo resultados satisfactorios. La funcionalidad de la aplicación obtuvo un porcentaje del 94% y la usabilidad un puntaje de 86.87. Esto indica que la aplicación web cumple con las expectativas de los usuarios, ya que es sencilla de utilizar sin necesidad de tener conocimientos técnicos. Por último, en las pruebas de cobertura realizadas se pudo comprobar el largo alcance de comunicación de los dispositivos LoRaWAN instalados.

Palabras clave: LoRaWAN, Monitoreo de Humedad de Suelo, Aplicación web, Agricultura de Precisión, Tecnología LPWAN.

ABSTRACT

The Internet of Things (IoT) applied to agriculture has allowed achieving technological development in this sector by using technologies such as LPWAN networks. These technologies enable obtaining a more accurate control of the information in real-time, thus reaching an improvement in the decision making in crop management to improve crop quality.

This work aims to apply LoRa technology in a strawberry crop to monitor soil moisture through a web application. The application allows the visualization of the data obtained from the sensors installed in the soil. Depending on the soil moisture levels, it displays messages to the user to know when the crop needs water.

The SCRUM methodology was used for the development of the web application. It allowed the participation of the stakeholders, obtaining constant feedback. For the coding, the Node JS, Express JS, and Semantic UI frameworks were used for the front-end and back-end development in conjunction with the Firebase platform as authentication and database manager.

At the end of the project, satisfactory results were found; the application's functionality obtained a percentage of 94%, and a score of 86.87 in usability. The results showed that the web application meets the users' expectations, as it is simple to use without technical knowledge. Finally, in the coverage tests performed, it was possible to verify the long communication range of the LoRaWAN devices installed.

Keywords: LoRaWAN, Soil Moisture Monitoring, Web Application, Precision Farming, LPWAN Technology.

CAPÍTULO 1. INTRODUCCIÓN

1.1 PLANTEAMIENTO DEL PROBLEMA

El sector agrícola representa un aporte importante a la economía del Ecuador, pues el 8% de su PIB corresponde a la producción agropecuaria. Siendo así que la mayoría de los alimentos que se consumen en el País vienen de dicha producción [1]. Para poder impulsar el desarrollo de este sector y llevarlo a otro nivel es primordial el uso de las últimas tecnologías del mercado. Pero como la gran parte de este sector está compuesto por pequeños y medianos agricultores que no cuentan con los recursos económicos suficientes para invertir en tecnologías costosas [2], será importante la búsqueda de soluciones que no impliquen altos costos [3].

El proceso de cultivos agrícolas se ve afectado por diversas variables físicas lo que muchas veces ocasiona la pérdida de los cultivos. Esto se da debido al control incorrecto de variables como la humedad, la temperatura, el pH del suelo entre otras [2]. La falta de control de estas variables debido a la extensión de los cultivos, junto con la escasa implementación tecnológica en el sector agrícola, provoca dificultades para los agricultores en el cultivo de sus productos [2][4].

En el caso específico de los cultivos de fresas, una de las principales variables que influye en su producción es la humedad del suelo, ya que de esta depende la maduración y engrose de las fresas. Un mal manejo de la humedad provoca diversos cambios e incluso enfermedades en el desarrollo de las plantas [5][6].

A fin de aportar en la solución de este problema, se plantea la aplicación de la tecnología LoRa en un cultivo de fresas ubicado en Tababela, sector nuevo aeropuerto de Quito, para realizar el monitoreo de la humedad del suelo y mediante el análisis de los datos obtenidos a partir de los sensores implementados, se pretende ayudar a obtener un proceso de cultivo más eficiente y a bajo costo. Esta solución permite no solo mejorar el control de la humedad del suelo sino también ayuda a gestionar de manera más eficiente los recursos naturales, en este caso el agua [4].

Mediante el uso del Internet de las Cosas (IoT, por sus siglas en Inglés) y la tecnología LoRa en la agricultura, se puede obtener procesos de cultivo mucho más eficientes y automatizados, logrando así un desarrollo tecnológico a bajo costo en el sector agrícola, evitando pérdidas de productos por las malas prácticas de producción [2][4].

1.2 JUSTIFICACIÓN PRÁCTICA

Gran parte del sector agrícola del País está conformado por pequeños agricultores de las zonas rurales, que no cuentan con presupuesto para implementar herramientas tecnológicas costosas en sus cultivos, por lo que prefieren evitar su uso. Sin embargo, en la actualidad existen soluciones a bajo costo que permiten mejorar sus prácticas de producción. Una de ellas es la aplicación de la tecnología LoRa en los cultivos [7], [8].

Como solución a este problema, el proyecto de Diseño de una red LoRaWAN en un cultivo de fresas para monitoreo de la humedad del suelo realizará una evaluación del comportamiento de esta tecnología en la parte práctica mediante la aplicación de LoRaWAN en un cultivo de fresas ubicado en la parroquia de Tababela perteneciente al Distrito Metropolitano de Quito. Esta propuesta analizará los datos obtenidos de los sensores y realizará el monitoreo de la humedad del suelo. De esta manera se podrá mantener control constante de la humedad reduciendo las pérdidas de los cultivos.

1.3 JUSTIFICACIÓN TEÓRICA

La mayoría de los agricultores realizan sus actividades de cultivos en base a sus conocimientos empíricos. Esto provoca que sus decisiones en cuanto al riego, fertilización y enfermedades sean incorrectas pues no se basan en datos actuales sino en información basada en la experiencia o en datos de cultivos anteriores, lo cual provoca cultivos ineficientes y crea una desventaja competitiva [7].

La aplicación de la tecnología LoRa en la agricultura ofrece grandes ventajas para el desarrollo de este sector. Debido a que LoRa permite la comunicación a larga distancia entre los equipos que participan en la red. Esto representa un bajo costo de implementación y consumo energético, entre otras cosas, por lo que puede ser usado por pequeños agricultores, así como también por las grandes industrias agrícolas sin necesidad de que esto represente un problema en su presupuesto [3].

1.4 OBJETIVOS

1.4.1 Objetivo General

Diseñar una red LoRaWAN para un cultivo de fresas que permita el monitoreo de humedad del suelo aplicado a un caso de estudio.

1.4.2 Objetivos Específicos

1. Identificar las necesidades de los agricultores encargados del cultivo de fresas.
2. Implementar la red LoRaWAN en el cultivo de fresas para el monitoreo de la humedad del suelo.
3. Desarrollar una aplicación web que permita visualizar los datos obtenidos de los sensores de humedad.
4. Establecer niveles de humedad del suelo que permitan conocer el momento adecuado para realizar el proceso de riego de agua en el cultivo de fresas.

CAPÍTULO 2. MARCO TEÓRICO

2.1 INTERNET DE LAS COSAS (IoT)

La tecnología ha avanzado en los últimos años considerablemente tanto así que ahora no solo las personas se encuentran conectadas entre si sino también las cosas. Diferentes dispositivos como teléfonos, autos, electrodomésticos e incluso gafas pueden conectarse a Internet para aprovechar sus beneficios. Esta etapa del Internet en la que se extiende hacia los objetos se conoce como Internet of Things (IoT) o en español el Internet de las Cosas [9].

Según IBSG (*Internet Business Solutions Group*) de Cisco [9], el Internet de las Cosas es "el punto en el tiempo en el que se conectaron a Internet más cosas u objetos que personas".

Como se menciona en [9] el Internet de las Cosas se refiere básicamente a la interconexión de objetos cotidianos en la red que se equipan con algún tipo de inteligencia. Esta interconexión les permite a los objetos compartir información que sirve para la toma de decisiones en tiempo real [9].

El Internet de las Cosas representa una opción para mejorar la productividad de las empresas en diferentes campos mediante una distribución de la red mucho más amplia extendiéndose a los objetos de la vida cotidiana. Este concepto se resume en varios elementos entre los que se tiene sensores, actuadores, controladores, procesos, usuarios y la conectividad [9].

Los sensores y actuadores envían y reciben información a través de la red. Los procesos o usuarios utilizan esta información para realizar diversas actividades como la monitorización, el control, la optimización y el proceso de toma de decisiones [9].

2.1.1 Arquitectura de Sistemas de IoT

La arquitectura de sistemas de IoT se forma básicamente por 4 capas [9]:

Arquitectura de IoT	
Capa de detección de objetos	Capa compuesta por los sensores que miden variables como movimiento, temperatura, humedad, viento, etc.
Capa de intercambio de datos	Capa que realiza la transmisión de los datos obtenidos desde los sensores a través de la red de nodos.
Capa de integración de la información	Capa que realiza el procesamiento de la información que a su vez se transforma en conocimiento de valor para los distintos usuarios y/o servicios de la red.
Capa de servicios de aplicación	Capa que ofrece los servicios de contenido a los usuarios finales una vez que se ha procesado la información.

Tabla 1. Arquitectura IoT de 4 capas.

2.1.2 Áreas de aplicación

El Internet de las Cosas tiene muchos usos en distintos ámbitos donde puede ser aplicado y en cada uno de ellos ofrece grandes beneficios. A continuación, se mencionan algunos [9]:

- En el hogar, actualmente se tiene el monitoreo y control, así como también la automatización de los distintos sistemas. Mediante IoT se puede tener un control de la mayoría de los objetos de una casa como las luces, las puertas, los electrodomésticos, etc. Con el fin de optimizar los recursos energéticos encendiendo o apagando equipos en base al tiempo o a datos obtenidos de sensores.
- El término "Ciudades Inteligentes" o *Smart Cities*, por sus siglas en inglés, está en auge en los países más desarrollados. En este campo existen grandes beneficios que brinda el Internet de las Cosas como por ejemplo el control y monitoreo del tráfico, la inspección de edificios e incluso la administración de servicios como el alumbrado público que se controla a través del uso de sensores.

- En el sector de la agricultura y ganadería se puede utilizar la tecnología para llevar un control más exacto de la producción. A través de la implementación de sensores se puede obtener datos en tiempo real acerca de las condiciones del suelo, del ambiente, y de los demás indicadores necesarios para el manejo de cultivos o de animales. Mediante esta información pueden realizar un proceso de toma de decisiones acorde a los datos obtenidos.

2.2 AGRICULTURA DE PRECISIÓN

Este término surge de la aplicación de las tecnologías de información y comunicación (TICs, por sus siglas en inglés) en el campo de la agricultura mejorando el proceso de toma de decisiones a través de la obtención de datos en tiempo real mediante el uso de sensores. Estas decisiones intervienen en el riego de agua o en la aplicación de insumos agrícolas como fungicidas, fertilizantes o semillas, pero en cantidades correctas. Las TICs en la agricultura no solo ayudan a medir las variables físicas de los cultivos sino también a la aplicación de mejores prácticas de gestión en este campo [10].

La agricultura de precisión se divide en 3 etapas: la recolección de datos, el análisis, procesamiento e interpretación de datos y la aplicación de insumos en base a la información procesada [10]. En cada una de las etapas se involucra la tecnología ya sea por medio de GPS (*Global Positioning System*), sensores, sistemas expertos o programas computacionales, entre otros. Esto con el fin de obtener grandes beneficios en el sector agrícola como la optimización de recursos, el ahorro de insumos y mano de obra mejorando la calidad de los cultivos [10], [11].

2.3 TECNOLOGÍA LORA

LoRa es una tecnología inalámbrica como WiFi, LTE, Bluetooth o SigFox, diseñada para largas distancias que utiliza una modulación en radiofrecuencia. Esta modulación se denomina CSS (*Chirp Spread Spectrum*) que tiene como principal característica la baja potencia y las comunicaciones de largo alcance [12].

La tecnología LoRa fue patentada por Semtech en el 2012 pero actualmente es administrada por LoRa Alliance. Esta tecnología pertenece a la capa física de la arquitectura de red creando los enlaces de comunicación de largo alcance entre los sensores y gateways que están conectados a la red [12], [13].

A continuación, se presentan un resumen de las características de LoRa [12], [13]:

Características	Detalle
Entidad Administradora	LoRa Alliance
Frecuencia	868/915MHz
Rango	De 10 a 20 km
Trasferencia de datos	255 bytes
Modulación	CSS (Chirp Spread Spectrum)
Consumo de energía	Hasta 10 años de duración de la batería.
Cifrado	AES128
Conexión	Punto a punto
Tolerancia a interferencias	Alta
Sensibilidad para recepción de datos	Alta (-168dB)

Tabla 2. Principales características de la Tecnología LoRa.

2.4 PROTOCOLO LORAWAN

LoRaWAN como se menciona en su página oficial "... es un protocolo de red de baja potencia y área amplia diseñado para conectar de manera inalámbrica a Internet, 'cosas' que funcionan con baterías, en redes regionales, nacionales o globales..." [14]. Este protocolo se encuentra definido por LoRa Alliance en su propia especificación [14], [15].

El protocolo LoRaWAN se caracteriza por transmitir datos a bajas velocidades, pero a grandes distancias, por lo que es ideal para el Internet de las Cosas. La baja velocidad de transmisión de los datos ayuda a maximizar el tiempo de duración de la batería de los dispositivos que forman la red lo cual diferencia a esta tecnología de otras [15].

2.4.1 Arquitectura LoRaWAN

Como se menciona en su documento de especificación, la arquitectura de red LoRaWAN "...generalmente se presenta en una topología de estrella en la que los gateways retransmiten mensajes entre dispositivos finales y un servidor central en el back-end..."

[15]. Los gateways se comunican con el servidor mediante conexiones IP mientras que los dispositivos lo hacen por la comunicación LoRa o FSK de uno o varios gateways [15]. En la Figura 1 se presenta la arquitectura LoRaWAN.

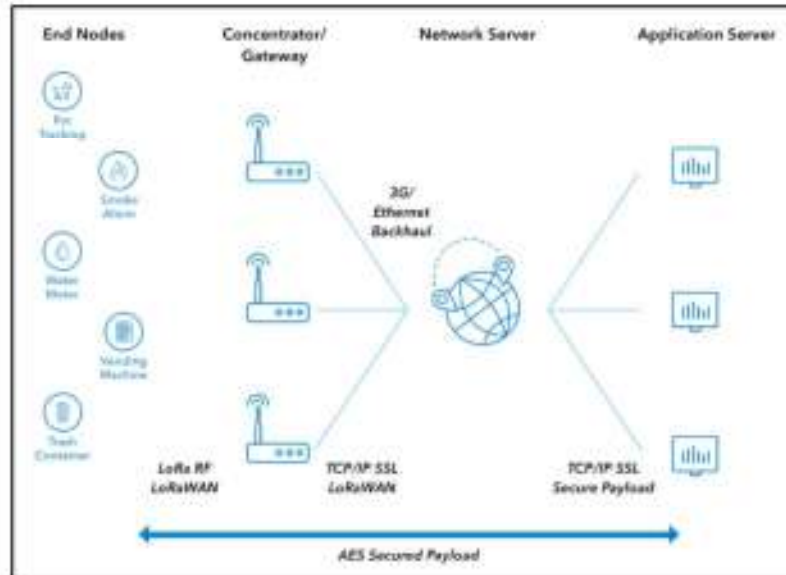


Figura 1. Arquitectura LoRaWAN. Fuente [16].

En esta arquitectura se pueden ver los principales elementos que conforman una red LoRaWAN [16]:

1. End Nodes: Los dispositivos finales son objetos que tienen un dispositivo de comunicación de baja potencia.
2. Gateways: Son los dispositivos que reciben y envían datos a los dispositivos finales.
3. Network Server: son servidores que direccionan los mensajes desde los dispositivos finales hacia el servidor de aplicación y viceversa.
4. Application Server: es el software que decide qué hacer con los datos recibidos desde los dispositivos finales.

2.4.2 Clases LoRaWAN

Todos los dispositivos LoRaWAN tienen al menos la funcionalidad de Clase A. Adicionalmente pueden implementar las opciones de Clase B y C, pero estas clases deben ser compatibles con la Clase A (Ver Figura 2).



Figura 2. Arquitectura en capas LoRaWAN. Fuente [17].

En la Tabla 3 se muestra las principales características de cada clase [15].

Clase A Bi-directional end-devices
<ol style="list-style-type: none"> 1. Comunicación bidireccional. 2. El intervalo de transmisión del end-device se basa en las necesidades de comunicación y una variación de tiempo aleatorio (tipo de protocolo ALOHA). 3. Sistema de end-device de menor potencia. 4. Para aplicaciones que requieren únicamente comunicación de enlace descendente desde el servidor.
Clase B Bi-directional end-devices with scheduled receive slots
<ol style="list-style-type: none"> 1. Permiten más ranuras de recepción. 2. Ventanas de recepción adicionales en horarios programados. 3. El servidor conoce cuando está escuchando el end-device.
Clase C Bi-directional end-devices with maximal receive slots
<ol style="list-style-type: none"> 1. Ventanas de recepción cerradas durante la transmisión. 2. Utilizan más energía. 3. Latencia más baja para comunicación entre servidor y end-device.

Tabla 3. Principales Características de las Clases LoRaWAN.

2.4.3 Activación de los dispositivos finales

Para formar parte de la red LoRaWAN, todos los dispositivos finales deben ser personalizados y activados. Para esto existen dos métodos, se puede hacer vía OTAA

(*Over-The-Air Activation*) o vía ABP (*Activation By Personalization*) [15]. Después de que los dispositivos son activados, en cada dispositivo se almacenan los siguientes datos:

Código	Nombre	Descripción
DevAddr	Device Address	Está formada por 32 bits que identifican al dispositivo final dentro de la red. Los 7 primeros bits son usados para el identificador de la red (NwkID). Este permite solucionar problemas de itinerancia y separar direcciones de red que se superponen. Los siguientes 25 bits corresponden a la dirección de red del dispositivo [15].
AppEUI	Application identifier	Es un ID de aplicación global en el espacio de direcciones de IEEE EUI64 que identifica a la entidad que procesa la solicitud de unión a la red (JoinReq) [15].
NwkSKey	Network Session Key	Es una clave de sesión de red específica para el dispositivo final. Es usada para verificar y calcular el MIC (<i>Message Integrity Code</i>) de los mensajes para garantizar la integridad de los datos [15].
AppSKey	Application Session Key	Es una clave de sesión de aplicación específica para el dispositivo final. Se utiliza para cifrar y descifrar el payload de los mensajes de la aplicación [15].

Tabla 4. Datos almacenados en el dispositivo final.

2.4.4 OTAA (*Over-The-Air Activation*)

En este método de conexión, todos los dispositivos finales deben seguir un proceso de unión antes de intercambiar datos con el servidor de red. Este proceso de unión necesita de la siguiente información para comenzar: el identificador del dispositivo final (DevEUI), el identificador de aplicación (AppEUI) y la clave de aplicación encriptada en AES-128 (AppKey). Esta clave es usada para generar las claves de sesión (AppSKey y NwkSKey) [15], [18].

El procedimiento de unión está formado por dos mensajes MAC, intercambiados entre el servidor de red y el dispositivo. Estos mensajes son un join request y un join accept. (Ver Figura 3).

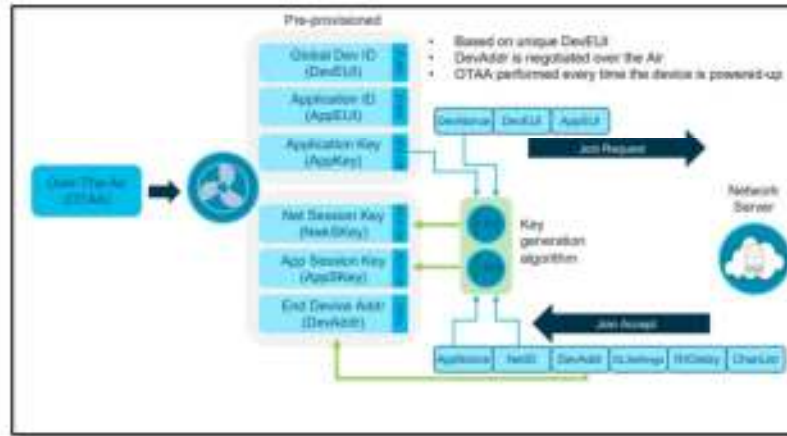


Figura 3. Método de Activación OTAA. Fuente [17].

Este procedimiento se inicia cuando el dispositivo final envía un mensaje join-request al servidor. Este mensaje contiene los siguientes campos [15]:

Campo	Tamaño (bytes)
AppEUI	8
DevEUI	8
DevNonce	2

Tabla 5. Campos del mensaje Join-Request. Fuente [15].

Mediante el valor DevNonce aleatorio, el servidor puede realizar un seguimiento de los valores de DevNonce que ya se han utilizado y puede ignorar solicitudes de conexión de dichos valores [15].

Si los datos enviados en el Join-Request son correctos, el servidor responderá con un Join-Accept para que el dispositivo pueda unirse a la red. Este mensaje contiene los siguientes campos [15]:

Campo	Tamaño (bytes)
AppNonce	3
NetID	3
DevAddr	4
DLSettings	1

RxDelay	1
CFList	(16) Opcional

Tabla 6. Campos del mensaje Join-Accept. Fuente [15].

En la Tabla 7, se muestra un resumen de los pasos realizados en el procedimiento de unión cada vez que el dispositivo se enciende de nuevo o se reinicia.

Procedimiento de unión con el método OTAA
1. El dispositivo final envía un join request a la red con su información de configuración.
2. El gateway recibe el request y lo envía al servidor de red.
3. El servidor de red verifica los datos enviados en el join request.
4. Si los datos son correctos, el servidor le crea una sesión temporal al dispositivo y la envía mediante el gateway. Si los datos son incorrectos rechaza la solicitud de unión.
5. El dispositivo final recibe la sesión y está listo para enviar datos a la red.

Tabla 7. Pasos de conexión con el método OTAA.

La ventaja principal de este método es la seguridad ya que se crea sesiones para los dispositivos cada vez que estos pierden conexión por lo que dificulta el robo o clonación del dispositivo [18].

2.4.5 ABP (Activation By Personalization)

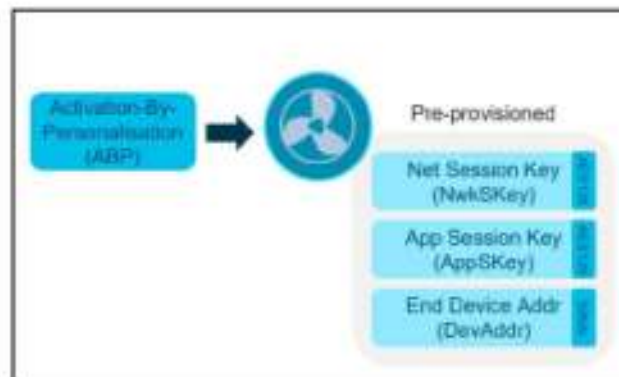


Figura 4. Método de Activación ABP. Fuente [17].

Este método de conexión vincula a un dispositivo final de manera directa a una red específica sin realizar el procedimiento de unión como el método OTAA. Esto lo realiza almacenando directamente los valores DevAddr, NwkSKey y AppSKey en el dispositivo final. Por lo tanto, se tiene la información necesaria para formar parte de una red específica cuando se inicia.

En la Tabla 8, se muestra un resumen de los pasos realizados en el procedimiento de unión cada vez que el dispositivo se enciende de nuevo o se reinicia.

Procedimiento de conexión con el método ABP
1. El dispositivo final envía los datos al gateway.
2. El gateway valida que los datos sean de la sesión.
3. Si los datos son correctos y corresponden a la sesión, se procesan. Caso contrario se rechazan.

Tabla 8. Pasos de conexión con el método ABP.

La ventaja principal de este método es evitar el procedimiento de unión a la red para enviar datos además la confirmación por parte del servidor ya no es necesaria ya que se han asignado manualmente las sesiones para los dispositivos finales [18]. Este método funciona en lugares que no tienen buena recepción o en dispositivos que están en movimiento pero su desventaja es la seguridad ya que las claves pueden ser clonadas o extraídas por un atacante [18].

CAPÍTULO 3. METODOLOGÍA Y TECNOLOGÍAS

3.1 Metodología

Para el desarrollo del sistema, se utilizará la metodología de desarrollo Scrum. Scrum es un marco de trabajo que reduce la complejidad del desarrollo de aplicaciones con el fin de satisfacer los requerimientos del cliente [8]. Se ha decidido usar esta metodología por su flexibilidad ante cambios y su entrega iterativa de software funcional. Lo cual va a permitir evaluar la aplicación web e implementar mejoras para una mejor visualización e interpretación de los datos obtenidos de los sensores de humedad.

En la primera fase, se realizará una identificación de las necesidades de los agricultores encargados del cultivo de fresas del caso de estudio. A partir de las necesidades encontradas se procederá a la fase 2, en donde se diseñará la red LoRaWAN para el caso de estudio planteado tomando en cuenta las dimensiones del terreno, la variable a medir, en este caso, la humedad del suelo y el protocolo LoRaWAN.

En una tercera fase, se elegirá el equipamiento necesario para la implementación de la red LoRaWAN en el cultivo de fresas, siguiendo el diseño de la fase anterior.

En la cuarta fase, se realizará la aplicación web que servirá de interfaz gráfica para que el usuario pueda visualizar los datos obtenidos de la red LoRaWAN implementada y los niveles de humedad establecidos que permitirán conocer el momento adecuado del riego de agua en el cultivo. Para el desarrollo de esta aplicación, se utilizará la metodología ágil Scrum. Esta metodología trabaja en iteraciones y parte de una lista de requerimientos del producto llamada Product Backlog. En base a esta metodología se realizarán las siguientes actividades en varias iteraciones:

1. Se parte del Product Backlog, que es una lista de requerimientos u objetivos del producto. Se definirán los requerimientos a implementarse en la aplicación web mediante la revisión de la literatura, la interpretación de los datos que se obtendrán de los sensores y la experiencia de los agricultores encargados del cultivo de fresas.
2. Como parte de una primera fase, se realizará el Sprint Planning en donde se planificará el trabajo a realizar y se asignarán las tareas de cada Sprint.
3. Se realizarán las tareas asignadas, siendo esta la etapa de codificación de los requerimientos de los interesados, para esto, se utilizará HTML, JavaScript y CSS.
4. En la siguiente fase se realizará el Sprint Review, en donde se presentará el producto desarrollado en el primer sprint. Como resultado de esto, se obtiene una lista de observaciones por parte de los interesados y el Scrum Master.
5. Como fase final, se entregará un avance que corresponde a un producto funcional.

En la última fase, se realizarán las pruebas de campo correspondientes con los sensores conectados a la red y a la aplicación web.

3.2 Tecnologías utilizadas

Para el desarrollo del presente proyecto se han utilizado las siguientes herramientas y tecnologías:

3.2.1 Protocolos

El término Internet de las Cosas o *Internet of Things* (IoT) en inglés, ha tenido un auge en los últimos años. IoT ha llegado a distintos sectores inclusive a la agricultura, surgiendo así la agricultura de precisión que se basa en el uso de sensores para controlar las variables físicas que intervienen en los cultivos.

Existen varias tecnologías LPWAN en el campo de IoT, como SigFox, LoRa, NBIoT, etc. Sin embargo, SigFox no es completamente abierto por lo que está restringido para ciertos

casos de uso y NBloT se basa en conexiones celulares por lo que requiere mayor consumo energético y altos costos de implementación [19].

La tecnología LPWAN que mejor se adapta al presente proyecto es LoRa mediante el protocolo LoRaWAN ya que permite el envío de datos a baja velocidad, pero a largo alcance por lo que la extensión de los cultivos no es un impedimento para su uso. Además, los dispositivos de LoRaWAN se caracterizan por sus baterías de larga duración [19] [15].

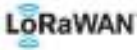
Nombre	Uso	Descripción	Logo
LoRaWAN	Diseño e implementación de la red LoRaWAN.	LoRaWAN es un protocolo de red de baja potencia y de área amplia que se diseñó para interconectar equipos de baterías de larga duración con Internet. Es utilizado para Internet of Things (IoT) [15].	


Tabla 9. Protocolo utilizado.

3.2.2 Diagramación y Prototipado

Es importante diseñar la arquitectura de un sistema antes de su construcción o implementación para tener una guía de lo que se quiere realizar y minimizar los errores en la práctica. Existen algunas herramientas que permiten realizar esto. Para el diseño de interfaces de aplicaciones se tiene Adobe XD que permite realizar diseños de interfaces similares a la realidad, para luego proceder a su implementación [20].

Para el diseño de diagramas se utilizó Draw.io y Lucidchart. Estas herramientas permiten la creación de gráficos y diagramas de todo tipo de manera sencilla para los usuarios. Fueron elegidas por ser las más conocidas en el mercado, permitiendo realizar las actividades en una cuenta gratuita [21] [22].

A continuación, se presentan estas herramientas.

Nombre	Uso	Descripción	Logo
Adobe XD	Diseño de Mockups para la Aplicación Web	Adobe XD es una herramienta que permite el diseño y creación de interfaces para aplicaciones web y móviles mediante la edición de gráficos	





		vectoriales y la experiencia de usuarios. Fue desarrollada por Adobe Inc [20].	
Draw.io	Diagramación de esquemas	Draw.io es un software que permite realizar diversos tipos de diagramas como UML, ER, diagramas de flujo, de red, entre otros [22].	
Lucidchart	Diagramación de esquemas	Lucidchart es una herramienta que permite crear gráficos de diferentes tipos para poder entender mejor las necesidades de equipos y procesos consiguiendo mejores decisiones en las diferentes áreas de negocios [21].	

Tabla 10. Herramientas de diagramación y prototipado utilizadas.

3.2.3 Lenguajes de Programación

Para implementar aplicaciones web, se tiene el lenguaje de programación HTML. Este lenguaje permite crear páginas web, y en conjunto con CSS y JavaScript las convierte en una aplicación web con funcionalidad avanzada.

HTML, CSS y JavaScript son los lenguajes base de programación al momento de crear aplicaciones web. Además, se tiene gran cantidad de frameworks y bibliotecas haciendo más sencillo su uso [23], [24], [25].

Nombre	Uso	Descripción	Logo
HTML	Desarrollo de Aplicaciones Web.	HTML (<i>HyperText Markup Language</i>) es la tecnología que se usa para definir una página web y su estructura. HTML usa un lenguaje de marcas para mostrar el contenido en el navegador web. Estos elementos pueden ser títulos, cabeceras, imágenes, videos, etc. [23].	
JavaScript	Desarrollo de Aplicaciones Web.	JavaScript es un lenguaje de programación que se creó para dar vida a una página web. Sirve para agregar interacción entre el usuario y el navegador web como por ejemplo,	


		agregar contenido HTML que modifique la página web, enviar solicitudes a servidores, entre otros [24].	
CSS	Desarrollo de Aplicaciones Web.	CSS (<i>Cascading Style Sheets</i>) es un lenguaje que añade estilo a una página web como el fondo, la fuente de texto, los colores, etc. A través del uso de CSS se puede definir la forma y presentación de los elementos de una página web [26], [25].	

Tabla 11. Lenguajes de programación utilizados.


3.2.4 Firebase

Firebase es una plataforma desarrollada por Google para la creación de aplicaciones de alta calidad de manera rápida y sencilla. Esta plataforma cuenta con varios servicios como base de datos, autenticación, mensajería, hosting, etc. Esto lo hace multiplataforma representando una gran ventaja frente a sus competidores [27].

Otra de las razones por las que se eligió Firebase es porque permite presentar a sus usuarios la posibilidad de crear aplicaciones sin necesidad de montar grandes infraestructuras para el despliegue de una aplicación [27].

Firebase a través del servicio Cloud Firestore presenta una base de datos No SQL que almacena los datos y permite consultar desde la aplicación de manera sencilla sin necesidad de crear APIs que controlen las consultas a la base de datos [28].

3.2.4.1 Base de datos

Nombre	Uso	Descripción	Logo
Cloud Firestore	Base de datos	Cloud Firestore es una base de datos NoSQL que permite gestionar los datos para aplicaciones web y móviles de manera sencilla. Ofrece los datos en tiempo real y sin conexión por lo que su capacidad de respuesta es independiente de la conectividad a Internet [28].	

		<p>Principales funciones [28]:</p> <ol style="list-style-type: none"> 1. <i>Flexibilidad:</i> permite tener estructuras de datos jerárquicas y flexibles almacenando los datos en documentos que se organizan en colecciones y subcolecciones. 2. <i>Consultas expresivas:</i> se pueden realizar consultas para obtener los documentos individuales o por colecciones dependiendo de los parámetros de búsqueda. Los datos se indexan de manera predeterminada por lo que el rendimiento de las consultas no depende del tamaño del conjunto de datos sino del tamaño del conjunto de resultados de la consulta. 3. <i>Actualizaciones en tiempo real:</i> Cloud Firestore actualiza los datos de cualquier dispositivo conectado mediante el uso de la sincronización de datos. 4. <i>Asistencia sin conexión:</i> Cloud Firestore permite que las aplicaciones puedan leer, escribir y consultar datos sin importar que el dispositivo no esté conectado a Internet ya que almacena los datos en caché. Una vez que el dispositivo tenga conexión, Cloud Firestore sincroniza los cambios locales pendientes. 	
--	--	--	--

Tabla 12. Base de datos utilizada.

Además, se tiene también el servicio de autenticación que permite la organización de usuarios en una aplicación mediante el uso de reglas que controlan las funciones de estos. Permite añadir a un proyecto la autenticación de usuarios de forma rápida y con unas pocas líneas de código [29].

3.2.4.2 Autenticación



Nombre	Uso	Descripción	Logo
Firebase Authentication	Autenticación de usuarios en la aplicación web.	Firebase Authentication permite conocer la identidad de los usuarios en las aplicaciones mediante los servicios backend que proporciona y bibliotecas UI preelaboradas mediante correo electrónico y contraseña, Google, Facebook, entre otros[29].	

Tabla 13. Herramienta de autenticación utilizada.

3.2.4.3 Despliegue

Se eligió la plataforma Heroku, por su manera rápida y sencilla de realizar el despliegue de la aplicación con solo usar un repositorio Git. Además de ser una de las plataformas más populares, ofrece también una versión gratuita y una prueba gratis que permite a los desarrollares conocer la plataforma [30].

Nombre	Uso	Descripción	Logo
Heroku	Despliegue de la aplicación web.	Heroku es una herramienta PaaS (Plataforma como Servicio) de computación en la nube. Es propiedad de Salesforce.com y fue desarrollada en 2007 iniciando con el lenguaje Ruby. Actualmente soporta varios lenguajes de programación [30]. Esta plataforma permite a los usuarios construir, administrar y monitorear aplicaciones de manera sencilla mediante el uso de un sistema de	

		<p>contenedores con servicio de datos integrado y una poderosa infraestructura [30].</p> <p>Estos contenedores se denominan dynos que son las partes fundamentales de la arquitectura de Heroku. Dynos son contenedores Linux virtualizados que fueron diseñados para ejecutar código mediante un comando específico dado por el usuario [31].</p>	
--	--	--	--

Tabla 14. Herramienta de despliegue utilizada.

3.2.5 Frameworks y Bibliotecas

Para la creación de interfaces se utilizó Semantic UI por sus componentes preconstruidos de HTML, CSS y JavaScript que hacen que las interfaces de usuario se vean amigables y al mismo tiempo fáciles de usar por cualquier usuario.

Semantic UI no es el framework más preferido por los desarrolladores sin embargo ha ganado popularidad gracias a su sencillez. Esto hace que disminuya el tiempo de desarrollo de las aplicaciones [32].


Nombre	Uso	Descripción	Logo
Semantic UI	Front-end de la Aplicación Web	Semantic UI es un framework de CSS que permite crear diseños de interfaces amigables para el usuario a través del uso de HTML, CSS y JavaScript mediante componentes prediseñados [32].	

Tabla 15. Frameworks para front-end utilizados.

Para el desarrollo del servidor, una opción sencilla es el uso de Node JS y Express JS. Estos frameworks permiten crear el ambiente necesario para levantar el servidor y poder cargar la aplicación web en el navegador, además de que cuenta con diversas bibliotecas para realizar el trabajo. A continuación, se describen sus características.

Nombre	Uso	Descripción	Logo
--------	-----	-------------	------




Node JS	Backend de la Aplicación Web	Node JS es un framework open source que ejecuta el código de JavaScript fuera del navegador. Está diseñado para crear aplicaciones escalables. Este framework junto con Express JS permite levantar un servidor web localmente [33].	
Express JS	Backend de la Aplicación Web	Express JS es un framework de Node.js para aplicaciones web que provee una serie de características robustas para aplicaciones web y móvil. Este framework permite, entre otras cosas, la creación de APIs para el desarrollo backend de las aplicaciones [34].	
TTN	Backend de la Aplicación Web	TTN es una biblioteca de Node JS que permite gestionar los dispositivos agregados al servidor TTN a través de una aplicación web [16].	

Tabla 16. Frameworks y bibliotecas para back-end utilizados.

Para permitir la comunicación entre el back-end y el front-end se utilizó la biblioteca socket.io.


Nombre	Uso	Descripción	Logo
Socket.io	Comunicación entre back-end y front-end	Socket.io es una biblioteca de JavaScript que permite la comunicación en tiempo real entre el servidor y el cliente (navegador), además de ser bidireccional y basada en eventos [35].	

Tabla 17. Biblioteca para comunicación entre back-end y front-end utilizada.

3.2.6 Control de Versiones

En la creación de aplicaciones ya sea web, móviles o de escritorio es necesario mantener un control de versiones del código. Esto permite el trabajo en equipo y la minimización de errores. Para esto existen varias herramientas que se basan en Git, como GitHub o BitBucket.

Se eligió BitBucket ya que se encuentra orientado hacia el trabajo en equipos. Permite también la integración con otras herramientas como Jira, Trello, etc. que ayudan a gestionar de mejor manera las tareas de un proyecto [36].

A continuación, se presenta una descripción de la herramienta seleccionada.


Nombre	Uso	Descripción	Logo
BitBucket	Software de Versionamiento	BitBucket es un servicio basado en la web que utiliza el sistema de control de versiones de Git y Mercurial. Está orientado a equipos profesionales ya que permite planificar y colaborar en proyectos así como también realizar las pruebas y el despliegue [36].	

Tabla 18. Control de versiones utilizado.

3.2.7 Entornos de Desarrollo

Uno de los pasos importantes en la codificación de las aplicaciones es la elección del IDE de desarrollo es decir la herramienta donde se va a escribir el código. La mayoría de estas herramientas vienen con plugins instalables para diferentes lenguajes lo que ayuda en la codificación. Pero sin duda una de las más sencillas y fáciles de instalar es Visual Studio Code [37].

A continuación, se presenta la descripción de la herramienta seleccionada.


Nombre	Uso	Descripción	Logo
Visual Studio Code	IDE de Desarrollo	Visual Studio Code es una herramienta que permite la edición de código fuente. Posee un conjunto de extensiones para el uso de diferentes lenguajes de programación como Java, JavaScript, Python, entre otros [37].	

Tabla 19. Entorno de desarrollo utilizado.

CAPÍTULO 4. DESARROLLO DE LA SOLUCIÓN

4.1 DESCRIPCIÓN DEL CASO DE ESTUDIO

Para el presente proyecto se ha seleccionado como caso de estudio un cultivo de fresas ubicado en la parroquia de Tababela perteneciente al Distrito Metropolitano de Quito. Este sector al ser rural se caracteriza por tener gran afluencia de agricultores dedicados al cultivo de varios productos como maíz, frejol, fresas, arveja, entre otros. En la Figura 5 se presenta el mapa del sector de ubicación del terreno.



Figura 5. Mapa de acceso al Cultivo de Fresas Renan Silva.

4.1.1 Descripción del cultivo de fresas

El cultivo de fresas pertenece al señor Renan Silva quien ha trabajado toda su vida en la siembra de productos como maíz, frejol y fresas. Este cultivo tiene una extensión de 6900 m² aproximadamente con un total de más o menos 134 camas y 60000 plantas de fresas.

La longitud de las camas de fresas es de 50 metros de distancia. Cada cama de fresas tiene 40cm de ancho y 40cm de alto. Se tiene dos filas de plantas en cada cama con una separación de 30 cm por planta. El sistema de riego empleado en el cultivo es por goteo siendo un sistema de cinta perforada que va debajo del plástico de cada cama.

En la Tabla 20 se muestra un resumen de las características del caso de estudio.

Característica	Valor
----------------	-------

Extensión del terreno	6.900 m ² aproximadamente
Tipo de cultivo	Cultivo de fresas
Variiedad de planta	Monterrey
Número de plantas	60.000
Número de camas	134
Tipo de suelo	Arenoso

Tabla 20. Características del caso de estudio.

El plano de terreno donde se encuentra ubicado el cultivo de fresas se observa en la Figura 6.

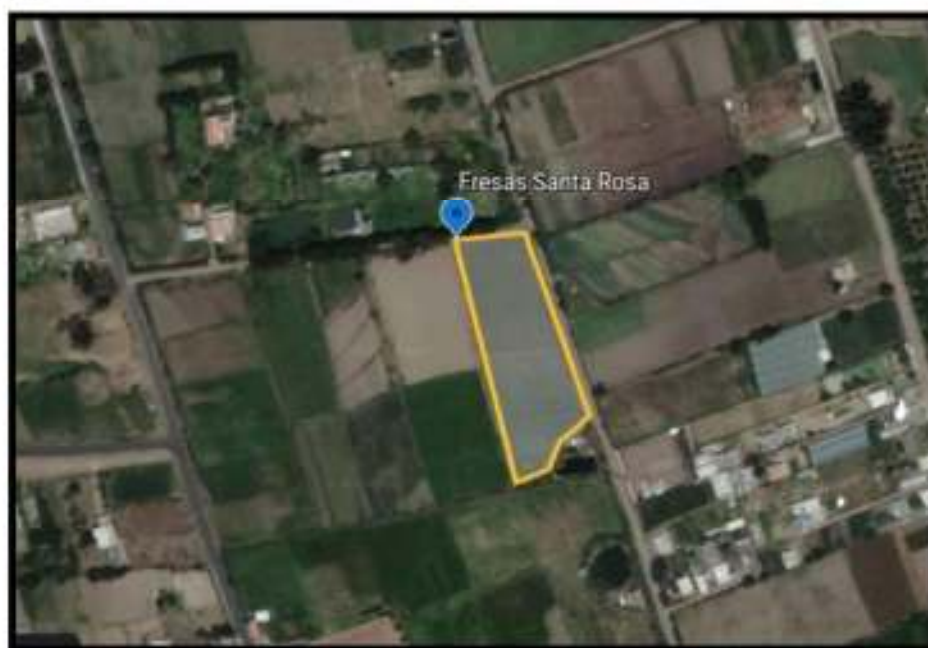


Figura 6. Plano de referencia del cultivo de fresas tomado de Google Earth. Fuente [38]

4.1.2 Proceso de cuidado del cultivo de fresas

La fresa es una planta herbácea de porte rastrero es decir sus ramas cuelgan al suelo y las frutas se desarrollan en contacto con la tierra. El cuidado de las fresas implica varios factores importantes entre ellos el riego, la fertilización, la poda y la fumigación [39].

El riego de las fresas es regular y se realiza dependiendo del clima, de la textura del suelo y necesidades de la planta. Para el caso del clima en el verano se requiere el riego todos los días, pero en invierno no es necesario ya que podría incrementar demasiado el nivel de humedad del suelo provocando enfermedades [40]. El proceso de riego es importante para mantener el suelo húmedo ya que el agua es uno de los factores más importantes en el cultivo de fresas tanto para su crecimiento cuando recién son plantadas como para la maduración de sus frutos [41].

Otro de los pasos a seguir en el cultivo de las fresas es la fertilización pues al igual que el agua se debe agregar fertilizantes todos los días. Estos fertilizantes dependerán del estado de la planta ya sea que esta necesite calcio, potasio, fósforo, zinc, entre otros nutrientes esenciales [40].

Finalmente otro de los pasos en el cultivo de las fresas es la poda, este paso se lo realiza dependiendo de la variedad de la planta y en cierto tiempo de su crecimiento, como se menciona en [42] durante la primera etapa de crecimiento se recomienda podar las flores cuando recién aparecen para promover el desarrollo de las raíces.

En todo el proceso de cultivo de las fresas existen varios requerimientos que son determinantes para obtener frutos sanos y de buena calidad. A continuación, se listan algunos de estos [39]:

1. Temperatura: según la experiencia de varios agricultores, aseguran que la temperatura es un factor primordial en este campo sin embargo en ciertos cultivos como en el de la fresa no es tan fácil de controlar ya que esto depende del clima que se tenga. En este País, el clima es muy variable por lo que representa el mayor problema en la agricultura. Por ejemplo, en el verano, se puede obtener la maduración temprana de los frutos, pero de un tamaño pequeño lo cual impide su comercio sin embargo en el invierno se puede tener enfermedades por la excesiva humedad del suelo. Los rangos óptimos oscilan entre los 15 y 20 grados centígrados.
2. Humedad: uno de los factores más importantes para el cultivo de la mayoría de los productos es la humedad. Desde el inicio de la siembra hasta el crecimiento de los frutos. El nivel de humedad interviene en el crecimiento de los frutos, así como también para la aparición de varias enfermedades por el manejo incorrecto de este factor [40]. Según se menciona en [39] la humedad debe estar entre un rango de 65 y 70%, caso contrario pueden aparecer ciertas enfermedades y afectar la

producción de los cultivos. Estos valores varían dependiendo del método de medición de humedad.

3. Luz: otro factor importante para la productividad es la luz por lo que se recomienda tener al menos 12 horas de luz diarias.
4. Suelo: uno de los factores primordiales para escoger el terreno donde sembrar es el tipo de suelo. Principalmente para el cultivo de fresa se recomienda los suelos de tipo arenosos con buena aireación y con un pH de 6-7. En [39] se mencionan los elementos que debe tener un suelo óptimo para un cultivo de fresas:
 - 50% de arena silíceo
 - 20% de arcilla
 - 15% de calizas
 - 5% de materia orgánicas

4.1.3 Descripción del problema

Como se ha mencionado en el apartado anterior, existen varios factores que intervienen en el proceso de cultivo de las fresas y de los cuales depende la obtención de frutos de buena calidad. Uno de los más importantes es el control de la humedad, siendo el riego de agua en las plantas, un paso principal para la buena productividad de las fresas.

Este paso depende en su mayoría del nivel de humedad que tiene el suelo, si este nivel es alto produce ciertas enfermedades ya sea por la pérdida de nutrientes o un crecimiento lento de las raíces. Si por el contrario la humedad es escasa se produce una reducción del tamaño de los frutos y se empeora su calidad por la aparición de enfermedades [40]. Estos problemas ocasionan grandes pérdidas para los agricultores si es que no se controlan a tiempo.

Una de las enfermedades que se produce por la humedad alta es la mancha púrpura o viruela del fresa, esta enfermedad provoca manchas en las hojas y si es que no se retiran las hojas dañadas a tiempo provocan daños en todo el cultivo. Sin embargo, para retirar estas hojas se requiere un proceso de poda y esto ocasiona que el agricultor tenga que invertir en mano de obra además de aplicar fungicidas para parar la enfermedad [43].

Así mismo en el caso de las frutas se produce la podredumbre gris o más conocida como botrytis. Esta enfermedad es causada por la humedad excesiva y aparece como manchas en los frutos. Para este caso se requiere retirar las fresas para evitar su propagación lo que provoca pérdidas en las cosechas [43].

Otra de las enfermedades que se presentan en el inicio de las plantas es la podredumbre del cuello o fitoftora, esta enfermedad produce un hongo que a su vez provoca la sequedad de la planta y la mejor solución para esto es remover la planta ocasionando pérdidas que el agricultor [43].

También existen varias enfermedades o plagas por la falta de humedad, como la araña roja o ácaros causada por la sequía es decir la falta de riego en el suelo. Esta plaga afecta a las hojas y posteriormente al crecimiento de la planta para esto es necesario la aplicación de fungicidas representando un aumento de gastos para el agricultor [43].

La humedad y la temperatura son los principales aspectos que se deben controlar en un cultivo de fresas. Sin embargo, la temperatura depende del clima por lo que resulta muy difícil de controlarla en un cultivo realizado en la tierra. Existen soluciones con las que se puede contrarrestar este problema, pero resultan bastante costosos para los pequeños agricultores.

Para el control de la humedad, por el contrario, se busca una solución óptima al mismo tiempo que pueda ser de un presupuesto considerable para el agricultor que no le ocasione mayores problemas de inversión y que incluya la tecnología para la optimización de procesos como se lo hace en otros campos.

Algunas de las herramientas tecnológicas representan altos costos en la mayoría de los casos y los pequeños y medianos agricultores no cuentan con los recursos de inversión necesarios para aplicar la tecnología en sus cultivos. Pero existen otras tecnologías como LoRaWAN que promete ser rentable para el agricultor.

4.2 DISEÑO DE LA RED LORAWAN

4.2.1 Diseño de la red LoRaWAN para el caso de estudio

En base a la descripción del caso de estudio se debe definir el número de sensores que serán necesarios para monitorear la humedad del suelo. Esta definición depende de varios factores como la variabilidad de la humedad del suelo, las características específicas del terreno como el área del campo, la precipitación, entre otros factores climáticos [44] [45].

Uno de los factores más importantes a definir es la homogeneidad del suelo. Es decir, si el suelo donde se van a colocar los sensores es homogéneo o heterogéneo. Según se menciona en [45] la mayoría de los suelos son heterogéneos cuando los campos son extensos, pero el presente caso de estudio es un campo pequeño por lo que la mayoría de

sus puntos son homogéneos y por ende el tiempo de riego se realiza igual en todo el terreno.

Otro de los factores a tomar en cuenta en la instalación de los sensores es la profundidad de ubicación. Esto depende del largo de las raíces de la planta. En el caso de estudio, las raíces miden aproximadamente 16 cm por lo que no es necesario colocar los sensores a diferentes profundidades pues la humedad no varía de manera considerable [46].

La ubicación de los sensores en el terreno también es un factor importante para considerar. Según se menciona en [44], los sensores se deben ubicar en lugares estratégicos del terreno, es decir, en los puntos donde haya mayor heterogeneidad de la humedad. Por lo cual se colocará uno por cada módulo de riego, en este caso el terreno se divide en dos partes para realizar el riego.

En base a las condiciones del terreno descritas anteriormente, la medición de campo y la revisión de la literatura realizada, se decidió usar 2 sensores para la red LoRaWAN, uno por cada módulo de riego.

Estos sensores se ubicarán a la profundidad de las raíces de la planta es decir a 16 cm por debajo de la cinta de goteo. El primer sensor se encargará de monitorear la mitad del terreno, y se ubicará hacia el medio de las camas de frutilla. El segundo sensor se ubicará en la otra mitad del cultivo, hacia el final de las camas de frutilla. Mediante esta ubicación de los sensores se obtendrá datos de las dos partes del terreno, es decir, del inicio y final de las camas.

En la Figura 7 se presenta un diagrama del diseño de la red LoRaWAN en el cultivo de fresas.

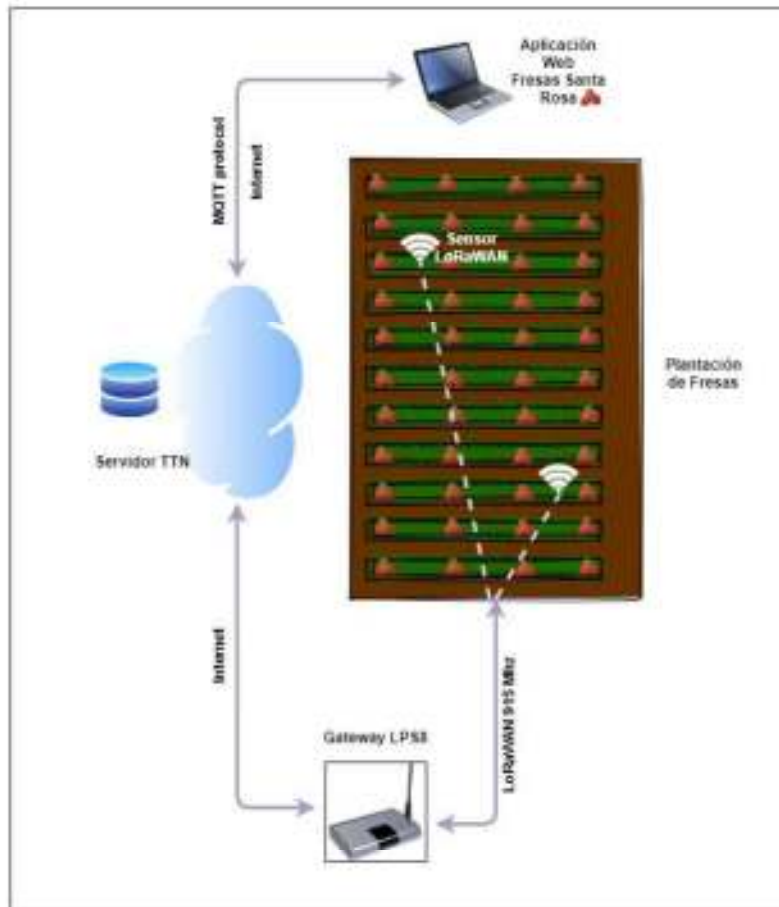


Figura 7. Diseño de la red LoRaWAN.

La red consta de dos sensores que se encuentran ubicados en lugares estratégicos del cultivo de fresas, estos sensores se conectan mediante wireless a la red LoRaWAN creada por el Gateway LPS8. Este dispositivo a su vez se conecta a un servidor LoRaWAN (TTN). Y a través del protocolo MQTT, los datos de los sensores son enviados al servidor para después ser mostrados en la aplicación web.

4.2.2 Equipamiento para la implementación de la red LoRaWAN

Para la red LoRaWAN, se eligió los equipos de la marca Dragino por ser open source tanto en hardware como en software además de contar con una amplia documentación.

Los equipos de Dragino cuentan con un software OpenWrt, que es una distribución del sistema operativo Linux para dispositivos embebidos. Es un sistema estable que cuenta con documentación y soporte necesario además de que el código fuente es open source. El hardware Dragino es un diseño de hardware abierto, es decir se puede modificar de manera libre según las condiciones lo requieran [47].

4.2.2.1 LoRaWAN Gateway

El gateway seleccionado es el "LPS8 Indoor LoRaWAN Gateway". LPS8 es un gateway open source que permite crear una red Wireless LoRa a una red IP mediante WiFi o Ethernet. Esta conexión permite enviar datos a distancias extremadamente largas pero a velocidades de datos bajas [48]. La Figura 8 muestra el gateway LPS8.



Figura 8. LPS8 Indoor LoRaWAN Gateway. Fuente [48].

En la Tabla 21 se presentan las especificaciones técnicas del producto [48].

Especificaciones técnicas del Gateway LPS8	
Hardware	
Procesador	400Mhz ar9331
RAM	64MB
Flash	16MB
Interfaz	
Puertos	1 puerto 10M/100M RJ45
WiFi	802.11 b/g/n
Power Input	5V DC, 2A, Type C
Conector USB	1 puerto USB 2.0
WiFi	
Especificación	IEEE 802.11 b/g/n
Banda de frecuencia	2.4 – 2.462 GHz

Tx power	<ol style="list-style-type: none"> 1. 11n tx power: mcs7/15: 2. 11db mcs0: 17db 3. 11b tx power: 18db 4. 11g 54M tx power: 12db 5. 11g 6M tx power: 18db
Sensibilidad WiFi	<ol style="list-style-type: none"> 1. 11g 54M: -71dbm 2. 11n 20M: -67dbm
LoRa	
<ol style="list-style-type: none"> 1. Sensibilidad de hasta -140dBm con SX1257 Tx/Rx front-end. 2. Rechazo de interferencia de 70 dB CW a 1 MHz offset. 3. Adaptación de velocidad de datos dinámica (DDR, <i>Dynamic Data-Rate</i>). 4. Interfaces frontales de radio Tx y Rx duales digitales. 	

Tabla 21. Especificaciones técnicas del Gateway LPS8. Fuente [48].

En la Tabla 22 se presentan las características y aplicaciones del gateway LPS8 [48].

Características Principales
<ol style="list-style-type: none"> 1. Sistema open source OpenWrt. 2. Administrado por Web GUI, SSH mediante WAN o WiFi. 3. Emula 49x demodulares LoRa. 4. Tiene 10 rutas de demodulación paralelas programables. 5. Se puede configurar y personalizar los parámetros regionales. 6. Admite inicio de sesión de niveles diferentes.
Aplicaciones
<ol style="list-style-type: none"> 1. Edificios inteligentes y domótica. 2. Logística y gestión de la cadena de suministro. 3. Agricultura inteligente. 4. Ciudades inteligentes. 5. Medición inteligente.

Tabla 22. Características y aplicaciones del Gateway LPS8. Fuente [48].

4.2.2.2 Sensor de humedad del suelo

El sensor de humedad de suelo seleccionado es el "LSE01 LoRaWAN Soil Moisture & EC Sensor" de la marca Dragino. LSE01 es un sensor LoRaWAN que mide la humedad, temperatura y conductividad del suelo conectando con el servidor LoRaWAN mediante wireless para el envío de los datos. Este sensor usa el método FDR (*Frequency Domain*

Reflectometry), es decir mide la constante dieléctrica del suelo y está calibrado para los suelos minerales que incluyen los tipos de suelo arenosos y arcillosos [49]. La Figura 9 muestra el sensor LSE01.



Figura 9. Sensor de humedad del suelo LSE01 LoRaWAN. Fuente [49].

En la Tabla 23 se presentan las especificaciones técnicas del producto [48].

Medida / Parámetro	Humedad	Conductividad	Temperatura
Rango	0-100.00%	0-20000uS/cm (25°C)(0-20.0EC)	-40.00°C – 85.00°C
Unidad	V%	uS/cm	°C
Resolución	0.01%	1 uS/cm	0.01°C
Exactitud	±3% (0-53%) ±5% (>53%)	2%FS	-10°C – 50°C: <0.6°C Otros: <0.6°C
Método de medida	FDR, con compensación de temperatura y conductividad de suelo.	Conductividad, con compensación de temperatura.	RTD y calibración.

Tabla 23. Especificaciones técnicas del Sensor LES01. Fuente [49].

En la Tabla 24 se presentan las características y aplicaciones del gateway LPS8 [48].

Características Principales
<ol style="list-style-type: none"> 1. Bajo consumo de energía. 2. Tiene una batería de 4000 mAh para un uso a largo plazo de hasta 10 años. 3. Monitoreo de la humedad de suelo. 4. Monitoreo de la temperatura de suelo. 5. Monitoreo de la conductividad de suelo. 6. Uplink periódicamente. 7. Para downlink se puede cambiar la configuración. 8. El sensor viene en una caja impermeable IP66.
Aplicaciones
<ol style="list-style-type: none"> 1. Agricultura inteligente.

Tabla 24. Características y aplicaciones del sensor LSE01.

El sensor LSE01 está configurado por default como LoRaWAN OTAA Class A por lo que cuenta con las claves de unión al servidor [49]. En la Figura 10 se muestra un diagrama de ejemplo del sensor en una red LoRaWAN.



Figura 10. Sensor LSE01 en una red LoRaWAN. Fuente [49].

4.3 IMPLEMENTACIÓN DE LA RED LORAWAN

4.3.1 Configuración e Instalación de equipos de la red LoRaWAN

4.3.1.1 Configuración del gateway LPS8

El dispositivo LPS8 viene con un cable USB del tipo C para encender y un puerto RJ45 para conectarse a una red. Al encender el dispositivo LPS8 se tiene 4 indicadores LED (Ver Tabla 25) como se muestra en la Figura 11.

LED	Icono	Color	Descripción
Power LED		Rojo	Indica que el dispositivo está correctamente encendido.




LoRa LED		Verde parpadeante	Indica que el módulo LoRaWAN se inició o está transmitiendo un paquete.
SYS LED		Azul sólido	Indica que el dispositivo tiene conexión a Internet, pero no tiene una conexión LoRaWAN.
		Azul parpadeante	Indica que el dispositivo está iniciándose por lo que parpadeará en azul por unos segundos y luego en rojo y azul en conjunto.
		Rojo	Indica que el dispositivo no tiene conexión a Internet.
ETH LED		Rojo	Indica el estado de la conexión Ethernet del dispositivo.

Tabla 25. Indicadores LED del dispositivo LPS8. Fuente [48].



Figura 11. Dispositivo LPS8 encendido.

El gateway LPS8 está configurado por default como un Wifi Access Point. Para configurar y acceder al dispositivo se debe conectar por vía Wifi o puerto Ethernet a una red de Internet. En este caso se conectó mediante Ethernet a una red de Internet y por Wifi desde la computadora para acceder a la configuración (Ver Figura 12).

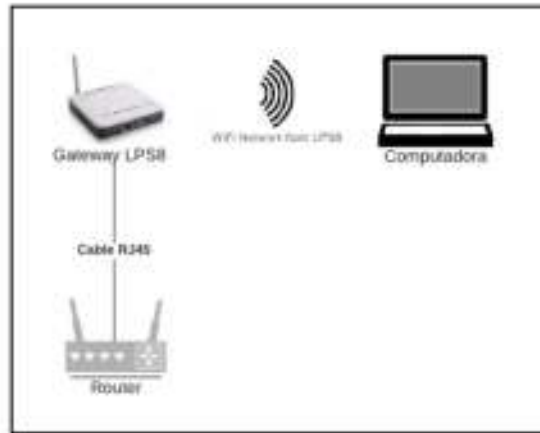


Figura 12. Diagrama de conexión del dispositivo LPS8.

Una vez que se enciende el dispositivo, este crea una red WiFi automáticamente como se muestra en la Figura 13. Esta red se crea con la información que se indica en la Tabla 26.

Nombre de la red	dragino-1eb7cc
Contraseña	dragino+dragino
Dirección IP	10.130.1.1

Tabla 26. Información de la red creada por el dispositivo LPS8.



Figura 13. Red Wifi creada por el dispositivo LPS8.

Una vez conectado a la red Wifi del dispositivo LPS8, se obtiene una dirección IP (<http://10.130.1.1/>) por medio de la cual se ingresa en el navegador a la interfaz de configuración del gateway. En la Figura 14 se muestra la interfaz de inicio de sesión del dispositivo LPS8. En la Tabla 27 se muestran las credenciales del dispositivo para iniciar sesión.

Nombre de usuario	root
Contraseña	dragino

Tabla 27. Credenciales del dispositivo LPS8

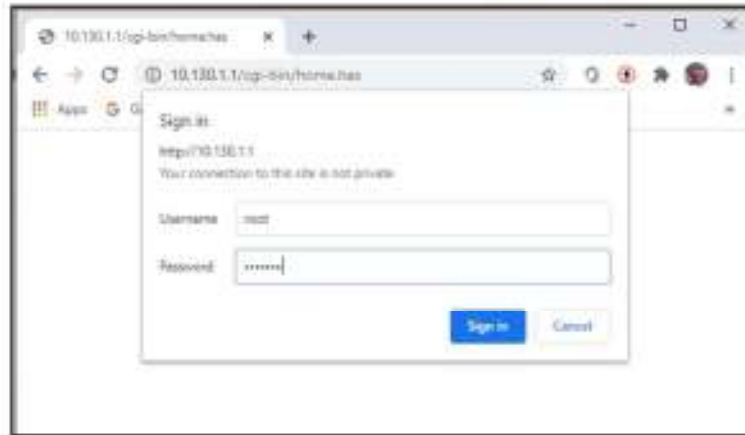


Figura 14. Interfaz de inicio de sesión para la configuración del dispositivo LPS8.

Cuando se ha iniciado sesión satisfactoriamente se muestra el portal del dispositivo donde se indica todas sus conexiones con sus respectivos indicadores de conexión. En este caso aún no se realiza la configuración de LoRaWAN por lo que se indica una marca x en el icono. (Ver Figura 15).



Figura 15. Portal de configuración del dispositivo LPS8.

Configuración de la red LoRaWAN:

Como primer paso se debe tener un *Gateway ID* del dispositivo. Esto se puede encontrar en la pestaña de configuración de LoRaWAN del dispositivo LPS8 como se muestra en la Figura 16. El gateway ID del dispositivo es *A840411EB7CC4150*.

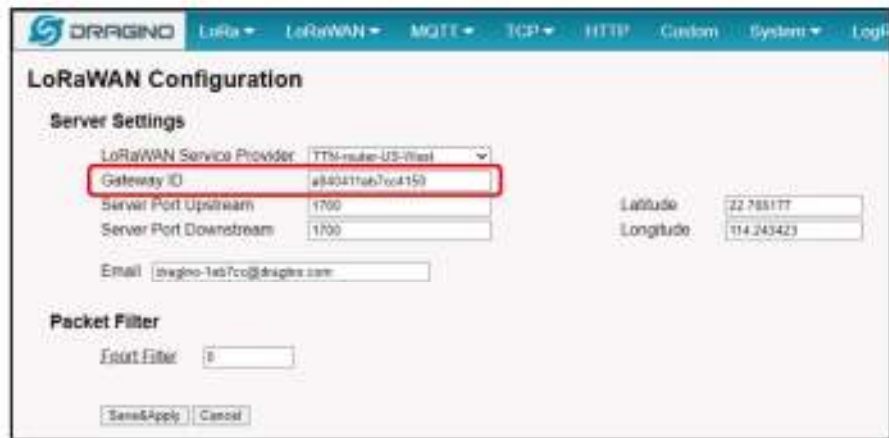


Figura 16. Interfaz de configuración de la red LoRaWAN.

Este dispositivo debe ser registrado en el servidor LoRaWAN TTN. Para esto se debe crear una cuenta en TTN e iniciar sesión (Ver Figura 17).



Figura 17. Página principal del servidor TTN.

A continuación, se debe ingresar en la consola de TTN para registrar los dispositivos. Se tiene dos opciones Applications y Gateways. En este punto se registró el gateway como se muestra en la Figura 18.



Figura 18. Registro del gateway en el servidor TTN.

En el registro del gateway se ingresa el gateway ID, la frecuencia, la región, la localización y el lugar de ubicación del gateway (Figura 19).



Figura 19. Resumen del gateway creado en el servidor TTN

Una vez configurado el gateway en el servidor TTN, en el portal de configuración se muestra la conexión de la red LoRaWAN del dispositivo LPS8. (Ver Figura 20)



Figura 20. Portal de configuración del dispositivo LPS8 después de la configuración.

En la Figura 21 se muestra los indicadores LED del dispositivo LPS8 después de haber configurado la red LoRaWAN.



Figura 21. Dispositivo LPS8 después de configurar la red LoRaWAN.

4.3.1.2 Configuración e instalación de los sensores LSE01

Estos dispositivos, así como el gateway deben ser registrados en el servidor LoRaWAN TTN. Para esto se tiene la opción de crear una aplicación en la cual se van a registrar todos los dispositivos de la red. En la Figura 22 se muestran los datos de configuración de la aplicación creada en el servidor TTN.

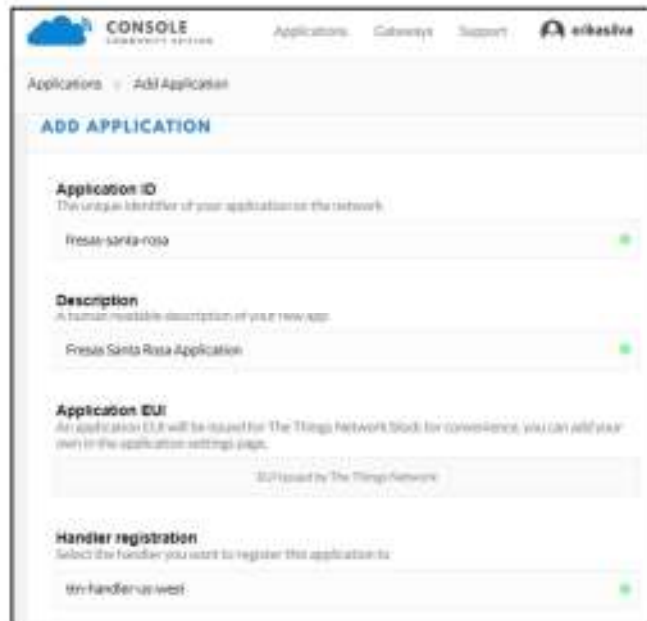


Figura 22. Creación de una aplicación en la consola del servidor TTN.

Dentro de la aplicación creada se deben agregar las APP EUIS de los dos sensores que van a ser registrados posteriormente de la siguiente manera:



Figura 23. App EUIS de los sensores LSE01-01 y LSE01-02.

Los sensores LSE01 están configurados para trabajar con el modo OTAA por default por lo que cada uno cuenta con sus claves de unión a la red. Después de haber creado la aplicación en la consola del servidor TTN, se deben registrar los dispositivos con sus respectivas claves.

Configuración del sensor LSE01-01:

Para registrar un dispositivo en el servidor TTN, se debe crear un Device en la aplicación creada, con las claves del equipo. En la Tabla 28 se muestran los datos del dispositivo LSE01-01 utilizados para realizar la configuración del servidor.

Información del Dispositivo LSE01-01	
DEVICE ID	LSE01-01
DEV ADDR	0182408A
DEV EUI	A84041A92182408A
APP EUI	33B29824F182408A
APP KEY	EA48F169846A5691182FCB9AC3144E65
APPSKEY	CD59612DA765227E272257ED26CDE441
NETSKEY	87CE21ECA6A488B6A6892E9C3EBFC44C

Tabla 28. Claves del dispositivo LSE01-01.

En la Figura 24 se muestra el registro del dispositivo en la consola del servidor TTN.

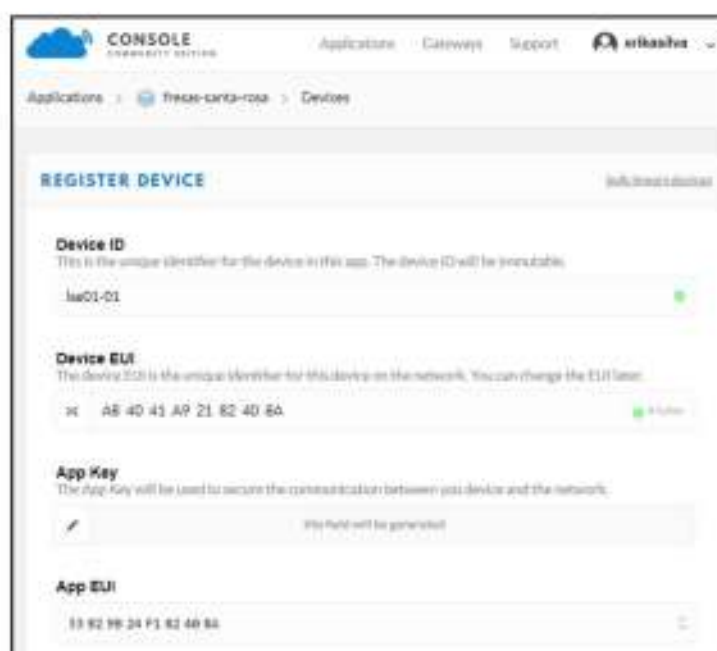


Figura 24. Registro del dispositivo LSE01-01.

Una vez registrado el dispositivo, se debe encender el sensor es decir colocar el Jumper de la placa en la posición FLASH. Después de encender el dispositivo, se debe unir a la red LoRaWAN automáticamente por el método de autenticación OTAA. En la Figura 25 se muestra el estado del dispositivo una vez conectado a la red.

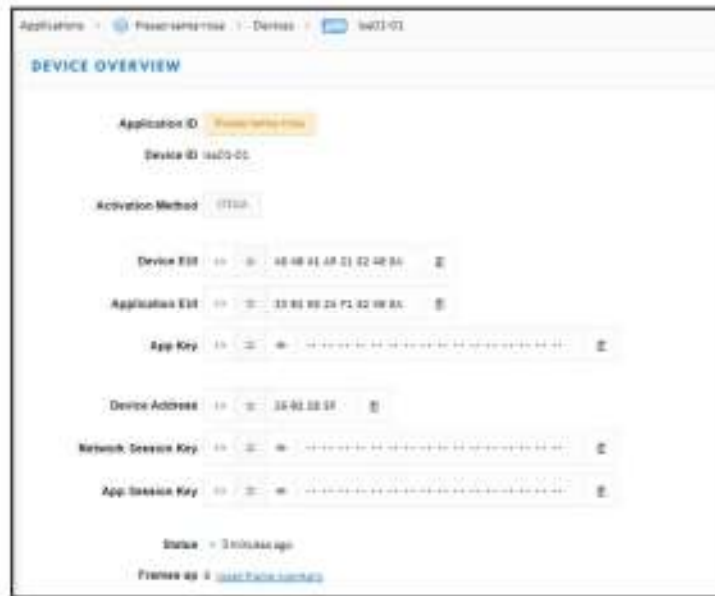


Figura 25. Estado del dispositivo LSE01-01.

El sensor envía a la red un payload de 11 bytes que se encuentran compuestos de la siguiente manera:

Tamaño (Bytes)	Valor
2	Voltaje de la batería
2	Temperatura
2	Humedad del suelo
2	Temperatura del suelo
2	Conductividad del suelo (EC)
1	Interruptor digital

Tabla 29. Componentes del uplink payload.

En la consola del servidor se debe decodificar los bytes del payload mediante un formato de payload preconfigurado como se muestra en la Figura 26.

```

1 function Decoder(bytes, port) {
2   // Decode an uplink message from a buffer [
3   var value=(bytes[0]<<8 | bytes[1]) & 0x0FFF;
4   var batV=value/1000; //Battery, units:V
5
6   value=bytes[2]<<8 | bytes[3];
7   if(bytes[4] & 0x00)
8     {value |= 0x0FFF0000;}
9   var temp_DS18B20=(value/10).toFixed(2); //DS18B20, temperature, units:Â°C
10
11  value=bytes[5]<<8 | bytes[6];
12  var water_SOIL=(value/100).toFixed(2); //water_SOIL, Humidity, units:%
13
14  value=bytes[7]<<8 | bytes[8];
15  var temp_SOIL;
16  if((value & 0x0000)>>15 == 0)
17    temp_SOIL=(value/100).toFixed(2); //temp_SOIL, temperature, units:Â°C
18  else if((value & 0x0000)>>15 == 1)
19    temp_SOIL=((value-0x0000)/100).toFixed(2);
20
21  value=bytes[9]<<8 | bytes[10];
22  var conduct_SOIL=(value/100).toFixed(2); //conduct_SOIL, conductivity, units:uS/cm
23
24  return {
25    bat:batV + " V",
26    tempC_DS18B20:temp_DS18B20+ " Â°C",
27    water_SOIL:water_SOIL+ " %",
28    temp_SOIL:temp_SOIL+ " Â°C",
29    conduct_SOIL:conduct_SOIL+ " uS/cm"
30  };
}

```

Figura 26. Formato de payload.

De esta manera, en la Figura 27 se pueden observar los mensajes enviados y recibidos entre el servidor y el sensor al momento del proceso de unión del dispositivo.



Figura 27. Mensajes intercambiados entre el sensor LSE01-01 y el servidor en el proceso de unión a la red.

Configuración del sensor LSE01-02:

Para configurar el sensor LSE01-02, se siguen los pasos anteriores. En la Tabla 30 se muestra la información utilizada para el registro del dispositivo en el servidor TTN.

Información del Dispositivo LSE01-02	
DEVICE ID	LSE01-02
DEV ADDR	0182408C
DEV EUI	A840419FA182408C
APP EUI	83B73EB1D182408C
APP KEY	2482485E12F99941568FEE8A36568F4D

APPSKEY	C9F2AE9A92AE4F565EB3B42C63187B9D
NETSKEY	A13E776E42DFA24358CF2D342DB5C938

Tabla 30. Claves del dispositivo LSE01-02.

En la Figura 28Figura 24 se muestra el registro del dispositivo en la consola del servidor TTN.



Figura 28. Registro del dispositivo LSE01-02.

Al igual que el dispositivo LSE01-01, se debe encender colocando el Jumper de la placa en la posición FLASH. Después de realizar este paso, el dispositivo se debe unir a la red LoRaWAN automáticamente. En la Figura 29 se muestra el estado del dispositivo una vez conectado a la red.

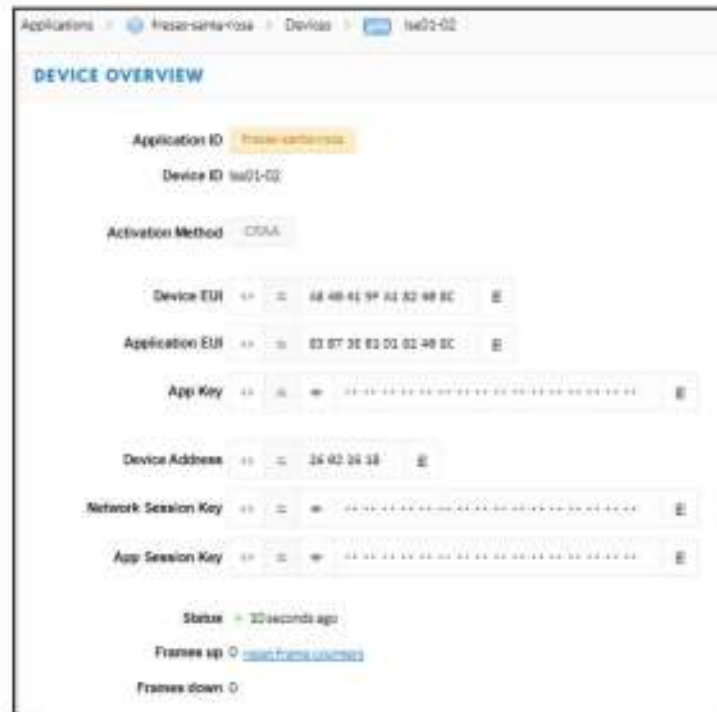


Figura 29. Estado del dispositivo LSE01-02.

De la misma manera después de encender el sensor, se puede visualizar el intercambio de mensajes en la consola del servidor durante el proceso de unión a la red (Ver Figura 30).



Figura 30. Mensajes intercambiados entre el sensor LSE01-02 y el servidor en el proceso de unión a la red

4.3.2 Comprobación del funcionamiento de los sensores instalados

Para motivos de prueba del funcionamiento, los sensores fueron instalados en una maceta con tierra ubicada cerca del gateway. En la Figura 31 y Figura 32 se puede mostrar los mensajes intercambiados entre los sensores y el servidor.

ID	sensor	payload
100001	1	2
100002	0	0
100003	0	2

Figura 31. Mensajes recibidos desde el sensor LSE01-01.

ID	sensor	payload
100004	1	2
100005	0	0
100006	0	2

Figura 32. Mensajes recibidos desde el sensor LSE01-02.

4.4 DESARROLLO DEL APLICATIVO WEB

4.4.1 Arquitectura de la aplicación

La aplicación web se ha desarrollado en tres componentes. El primer componente es la interfaz de la aplicación con la cual el usuario podrá interactuar, consta del front-end que es implementado usando HTML, JavaScript y CSS, además del uso del framework de Semantic UI para el diseño de las interfaces.

El segundo componente es el servidor de back-end utilizando los frameworks de Node JS y Express JS. El cual permite realizar la conexión de la aplicación con el servidor LoRaWAN TTN para el envío y recepción de datos.

El tercer componente es Firebase, que cumple las funciones de autenticación y base de datos. Permite la autenticación de los usuarios en la aplicación, el almacenamiento de los datos y la comunicación entre el front-end y la base de datos [28], [29].

El sistema será desplegado mediante el uso de Heroku. Este servicio ofrece la infraestructura necesaria para implementar y administrar aplicaciones web modernas basada en el uso de contenedores permitiendo el acceso de los usuarios desde cualquier parte del mundo [30].

En la Figura 33 se presenta el diseño de la arquitectura de la aplicación web.

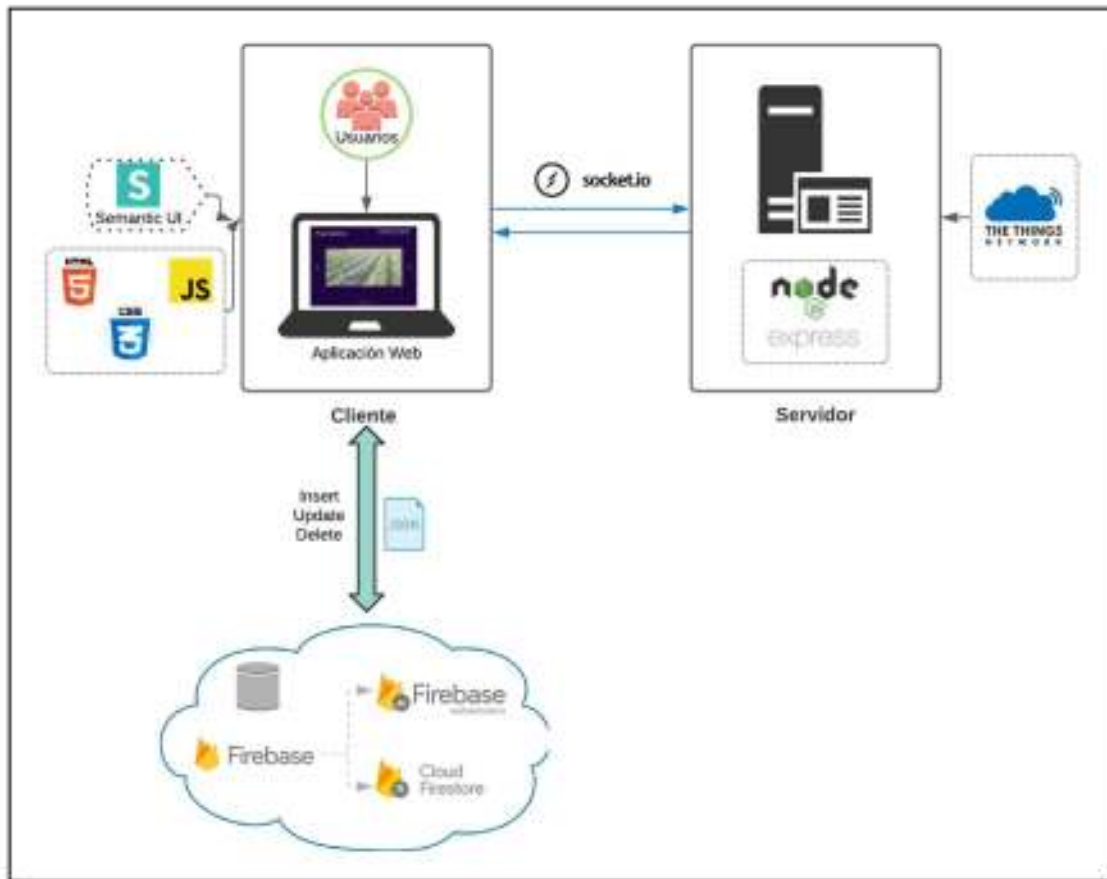


Figura 33. Arquitectura de la aplicación web.

4.4.2 Product Backlog

El desarrollo de la aplicación web se inicia con el product backlog que contiene la lista de requerimientos funcionales y la prioridad clasificada en Alta, Media, Baja de cada uno de los requerimientos. La Tabla 31 muestra los requerimientos de la aplicación web.

Código	Nombre	Descripción	Prioridad
R01	Creación de cuentas para el acceso a la aplicación web	El usuario podrá crear una cuenta con nombres, apellidos, correo electrónico y contraseña para ingresar a la aplicación web.	Alta
R02	Inicio de sesión en la aplicación web	El usuario podrá ingresar a la aplicación web para ver la información de sus cultivos.	Alta

R03	Registro de información de cultivos.	El usuario podrá registrar los siguientes datos de sus cultivos: <ol style="list-style-type: none"> 1. Nombre del cultivo, 2. Tipo de planta, 3. Variedad, 4. Cantidad de plantas sembradas, 5. Extensión del terreno, 6. Ubicación, 7. Observaciones. 	Alta
R04	Visualización de información de los cultivos,	El usuario podrá visualizar la información registrada de sus cultivos.	Alta
R05	Edición de la información de los cultivos,	El usuario podrá editar la información de los cultivos.	Media
R06	Visualización de los datos de los sensores.	El usuario podrá observar un panel con los valores principales de los sensores como la humedad, la temperatura, la conductividad del suelo, la batería y la fecha de la última lectura de datos del sensor en la pantalla principal de la aplicación.	Alta
R07	Notificación de riego en el cultivo.	El usuario será notificado en la pantalla principal a través de un cuadro de mensaje cuando la humedad de suelo esté fuera del rango establecido.	Alta
R08	Visualización de los sensores conectados en la aplicación	El usuario podrá visualizar los sensores conectados a la aplicación web mediante su ID de dispositivo.	Media
R09	Buscar registros de los datos de sensores por fecha.	El usuario podrá buscar los registros de los valores de los sensores según la fecha indicada.	Media

R10	Eliminación de registros de datos de los sensores.	El usuario podrá eliminar los registros de valores de sensores menos relevantes.	Media
R11	Registro de información de riegos.	El usuario podrá ingresar la siguiente información de los riegos realizados en el cultivo de fresas: 1. Fecha 2. Hora 3. Duración 4. Humedad Actual 5. Observaciones	Alta
R12	Visualización de información de riegos.	El usuario podrá visualizar la información registrada de los riegos.	Alta
R13	Edición de información de riegos.	El usuario podrá editar los registros de riego ingresados en la aplicación web.	Media
R14	Eliminación de registros de riego.	El usuario podrá eliminar los registros ingresados en la aplicación web que sean irrelevantes.	Media
R15	Buscar registros de riego por fecha.	El usuario podrá buscar los registros de riego ingresados por la fecha.	Media
R16	Restablecimiento de contraseña.	El usuario podrá cambiar su contraseña para ingresar a la aplicación web.	Baja
R17	Visualización de la pantalla de inicio de la aplicación web.	El usuario podrá visualizar en la pantalla de inicio de la aplicación web una secuencia de imágenes del cultivo de fresas.	Media

Tabla 31. Product Backlog de la Aplicación Web.

Después de haber definido los requerimientos iniciales de la aplicación web, se realiza la ejecución de los sprint. Estos sprint tendrán las fases de planificación, codificación, revisión y despliegue.

4.4.3 Sprint 1

4.4.3.1 Sprint Planning

Código	Nombre	Descripción	Prioridad	Story Points
R01	Registro de usuarios en la aplicación web	El usuario podrá crear una cuenta con nombres, apellidos, correo electrónico y contraseña para ingresar al aplicativo web.	Alta	31
R02	Inicio de sesión en la aplicación web	El usuario podrá ingresar a la aplicación web para ver la información de sus cultivos.	Alta	22
Total				53

Tabla 32. Sprint Planning - Requerimientos para el Sprint 1.

A continuación, se presentan cada uno de los requerimientos con sus subtareas a realizarse en el Sprint 1, en conjunto con los story points de cada una.

Requerimiento	Subtareas	Story Points
R01 - Registro de usuarios en la aplicación web.	1. Diseño e implementación de la estructura para la base de datos de la aplicación web.	8
	2. Integración de la aplicación con Firebase Authentication para la autenticación de usuarios.	3
	3. Integración de la aplicación con Cloud Firestore para el almacenamiento de los datos de la aplicación web.	3
	4. Diseño e implementación de la interfaz para crear la cuenta.	8
	5. Validar los campos Nombres, Apellidos, Correo Electrónico y Contraseña como obligatorios.	3
	6. Validar la contraseña ingresada con un mínimo de 6 caracteres.	3

	7. Validar el correo electrónico con un formato válido, es decir, debe contener los caracteres @ y ".".	3
R02 - Inicio de sesión en la aplicación web	1. Diseño e implementación de la interfaz de inicio de sesión.	8
	2. Validar campos vacíos en el inicio de sesión.	3
	3. Mostrar mensajes de error en el caso de que los datos ingresados en la interfaz de inicio de sesión estén incorrectos.	3
	4. Diseño e implementación de la interfaz de inicio después de haber iniciado sesión.	8
Total		53

Tabla 33. Tareas del Sprint 1.

4.4.3.2 Codificación

A continuación, se detalla la implementación de cada una de las tareas realizadas en el Sprint 1.

Requisito R01: Registro de usuarios en la aplicación web.

Para la implementación del registro de usuarios, primero se debe definir la estructura del almacenamiento de los datos en la base de datos.

Cloud Firestore permite organizar los datos en documentos y estos a su vez en colecciones y subcolecciones. Cada documento guarda los datos en campos a los cuales se les asignan valores, estos documentos se almacenan en colecciones lo que hace posible la ejecución de consultas de manera sencilla.

En la Figura 34, se presenta la estructura NoSQL diseñada para almacenar la información del proyecto en Cloud Firestore.

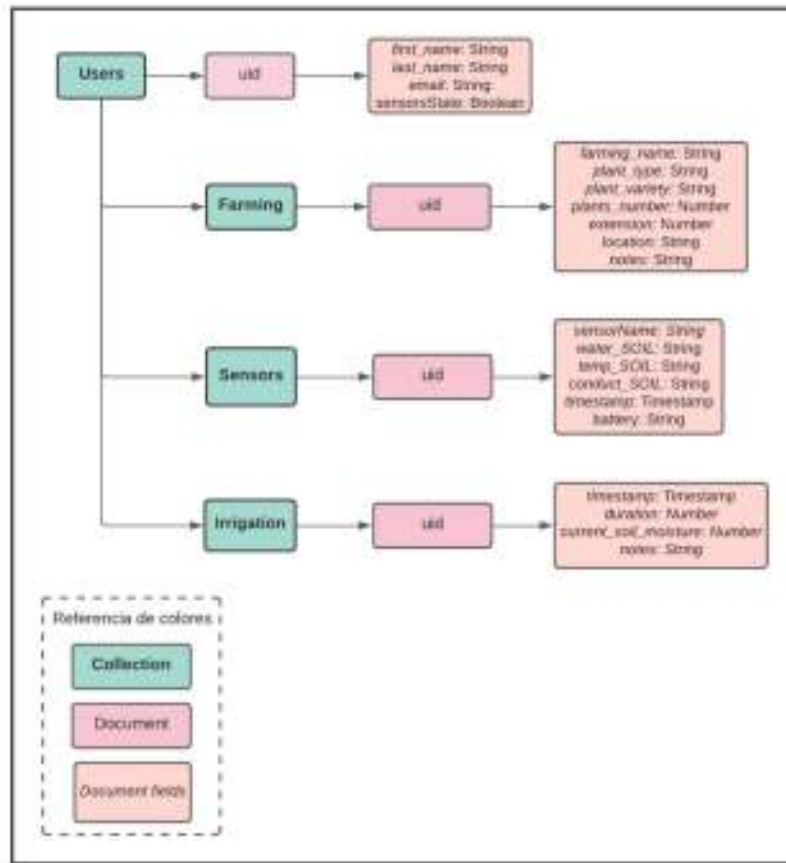


Figura 34. Estructura NoSQL de los datos en Cloud Firestore.

Integración de la aplicación con Firebase Authentication y Cloud Firestore:

Para poder usar los servicios de Firebase en la aplicación web, es necesario tener una cuenta de Google. Una vez que se tenga acceso a Firebase se puede acceder a la consola desde la cual se administran los proyectos creados. En la Figura 35, se muestran los proyectos creados para un usuario.

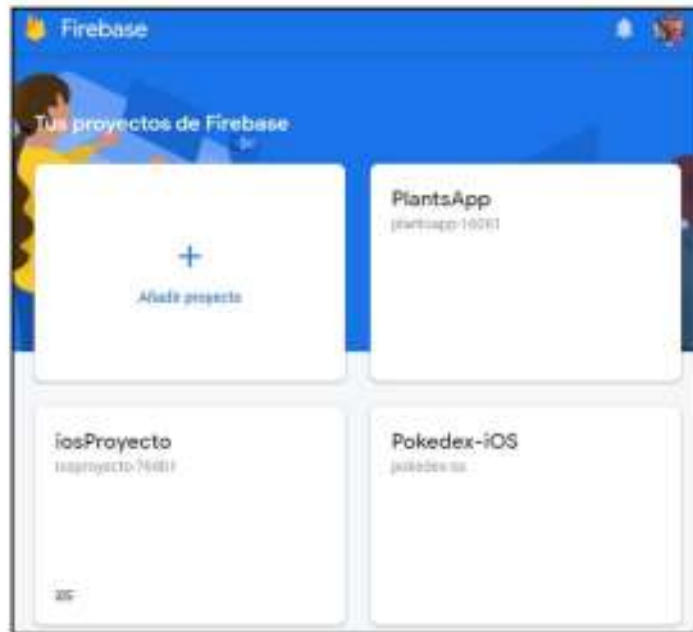


Figura 35. Proyectos creados en Firebase.

Para crear un nuevo proyecto en Firebase, se debe seleccionar la opción *Añadir Proyecto* y se desplegará una pantalla donde se pide ingresar el nombre del nuevo proyecto (Ver Figura 36).



Figura 36. Asignación de nombre al nuevo proyecto

Los siguientes pasos son la configuración de Google Analytics. Esta opción es recomendación de Google por lo que se seleccionará por default para el proyecto. Una vez

creado el proyecto, se abrirá la consola de Firebase en la cual se mostrará todos los servicios que pueden ser usados, como se muestra en la Figura 37.



Figura 37. Consola de Firebase para el proyecto Fresas Santa Rosa.

En la consola se realiza la integración de Firebase con el aplicativo web. Para esto se escoge la plataforma a usarse (Figura 38).



Figura 38. Plataformas disponibles.

Para añadir una aplicación web a Firebase se deben seguir los pasos que se muestran en la Figura 39.



Figura 39. Pasos para agregar una aplicación web en Firebase.

Para el paso 1 (Registrar la aplicación), se desplegará la pantalla que se muestra en la Figura 40. Esta pantalla solicita el nombre de la aplicación web.

La interfaz de usuario muestra el título 'Añadir Firebase a tu aplicación web'. El primer paso es 'Registrar la aplicación'. Hay un campo de texto con el nombre 'FresasSantaRosa'. Una opción seleccionada es 'Configura también Firebase Hosting en esta aplicación'. Hay un campo para el nombre de dominio 'fresas-santa-rosa' y un botón 'Registrar aplicación'.

Figura 40. Paso 1: Registro de la aplicación web en Firebase.

En el paso 2 (Añadir SDK de Firebase), se solicita añadir en la aplicación web, los SDK de todos los productos de Firebase que se requieran (Ver Figura 41).



Figura 41. Paso 2: Adición de SDK de Firebase.

En la Figura 42 se muestran los SDK añadidos en la aplicación web.

```
<script defer="" src="https://www.gstatic.com/firebasejs/7.15.5/firebase-app.js"></script>
<script defer="" src="https://www.gstatic.com/firebasejs/7.15.5/firebase-auth.js"></script>
<script defer="" src="https://www.gstatic.com/firebasejs/7.15.5/firebase-firestore.js"></script>
<script defer="" src="https://www.gstatic.com/firebasejs/7.15.5/firebase-init.js"></script>
```

Figura 42. SDK añadidos al proyecto.

El archivo *init.js* añadido en conjunto con los SDK contiene la configuración del proyecto creado en Firebase. La Figura 43 muestra el contenido de este archivo.

```
init.js
public > model > firebase > init.js >
1 const firebaseConfig = {
2   apiKey: "AIzaSy03590ca77PpmZ83EurS_K7xPmM3vSRaw",
3   authDomain: "fresasantarosa-938c6.firebaseio.com",
4   databaseURL: "https://fresasantarosa-938c6.firebaseio.com",
5   projectId: "fresasantarosa-938c6",
6   storageBucket: "fresasantarosa-938c6.appspot.com",
7   messagingSenderId: "612958611374",
8   appId: "1:612958611374:web:83b0fc7bb12b1532ec8a3f"
9 };
10
11 // Initialize Firebase
12 firebase.initializeApp(firebaseConfig);
13 var db = firebase.firestore();
```

Figura 43. Contenido del archivo *init.js*

En el paso 3 (Instalar CLI de Firebase), se solicita instalar la CLI de Firebase mediante el uso de Node JS (npm) con el comando: `npm install -g firebase-tools`. (Ver Figura 44 y Figura 45)



Figura 44. Paso 3: Instalación de CLI de Firebase.



Figura 45. Instalación de Firebase Tools.

Una vez realizada la integración de Firebase Authentication y Cloud Firestore con la aplicación web, se implementó la creación de usuarios.

En la Figura 46 se muestra la interfaz de creación de cuenta implementada con los campos Nombres, Apellidos, Correo Electrónico y Contraseña.

Figura 46. Requisito R01 – Pantalla de creación de cuentas.

De la misma manera, en la Figura 47 se muestra la validación de campos obligatorios, así como también la validación del correo electrónico y la contraseña.

Figura 47. Requisito R01 – Validación de campos.

Requisito R02: Inicio de sesión en la aplicación web.

En la Figura 48 se muestra la pantalla implementada para el inicio de sesión de la aplicación web.



The image shows a login form for 'Fresas Santa Rosa'. At the top, the brand name 'Fresas Santa Rosa' is displayed in green, with a cluster of strawberries to the right. Below the name is the title 'Iniciar Sesión'. The form contains two input fields: 'Correo Electrónico' and 'Contraseña'. A dark blue button labeled 'Iniciar Sesión' is positioned below the fields. Underneath the button is a link that says '¿Olvidaste tu Contraseña?'. At the bottom of the form is a light blue button that says '¿No tienes una cuenta? Regístrate'.

Figura 48. Requisito R02 – Pantalla de Inicio de sesión.

En el inicio de sesión se realiza la validación de las credenciales. Ya sea por los campos vacíos o porque el usuario no se encuentra registrado, se mostrará un mensaje de error informando sobre las credenciales incorrectas. En la Figura 49 se muestra el inicio de sesión con el mensaje de error por credenciales incorrectas.

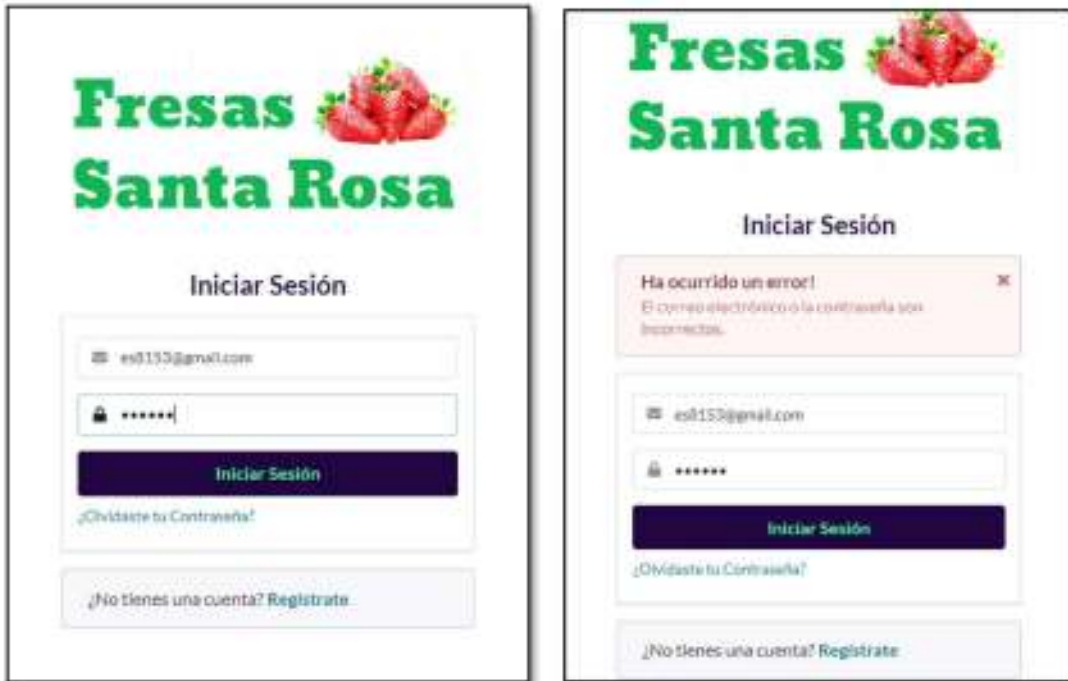


Figura 49. Requisito R02 – Inicio de sesión con credenciales incorrectas.

De la misma manera, en la Figura 50 y Figura 51 se muestra el caso exitoso del inicio de sesión. Si el usuario ingresa las credenciales correctas, podrá ingresar al sistema para visualizar su información.



Figura 50. Requisito R02 – Inicio de sesión con credenciales correctas.



Figura 51. Pantalla principal de la aplicación.

Por último, en la Figura 52 se muestra el registro de usuarios en la base de datos.

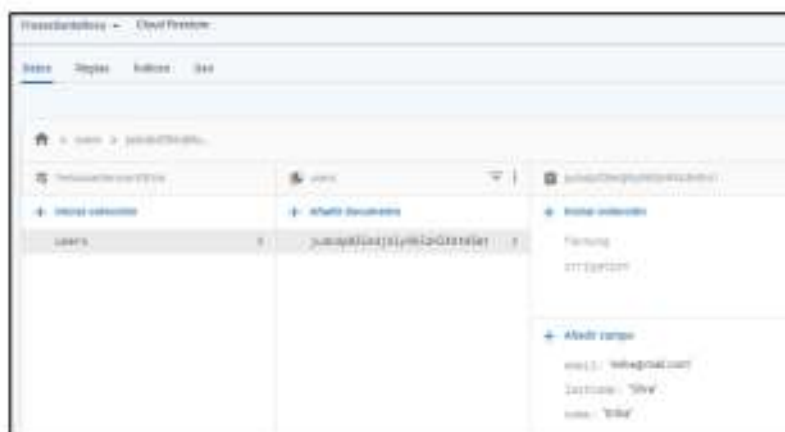


Figura 52. Registro de usuarios en la base de datos.

4.4.3.3 Sprint Review

Una vez completada la implementación de las subtarefas de cada requisito del sprint 1, se presenta el producto funcional para obtener las observaciones y recomendaciones pertinentes. (Ver Tabla 34).

Código	Nombre	Observaciones	Puntos estimados	Puntos finalizados	Puntos pendientes
--------	--------	---------------	------------------	--------------------	-------------------

R01	Registro de usuarios en la aplicación web	Ninguna	31	31	0
R02	Inicio de sesión en la aplicación web	Ninguna	22	22	0
Total			53	53	0

Tabla 34. Sprint Review del Sprint 1.

4.4.3.4 Despliegue

En el sprint 1 se realiza el despliegue de la aplicación mediante un servidor local usando los frameworks de Node JS y Express JS, para poder visualizar las funcionalidades implementadas. La ruta en la que se creó el servidor localmente fue <https://localhost:3000>.

4.4.4 Sprint 2

4.4.4.1 Sprint Planning

Código	Nombre	Descripción	Prioridad	Story Points
R03	Registro de información de cultivos.	El usuario podrá registrar los siguientes datos de su cultivo: <ol style="list-style-type: none"> 1. Nombre del cultivo, 2. Tipo de planta, 3. Variedad, 4. Cantidad de plantas sembradas, 5. Extensión del terreno, 6. Ubicación, 7. Observaciones. 	Alta	19
R04	Visualización de información de los cultivos.	El usuario podrá visualizar la información registrada de sus cultivos.	Alta	8

R05	Edición de la información de los cultivos.	El usuario podrá editar la información de los cultivos.	Media	14
R06	Visualización de los valores de humedad del suelo.	El usuario podrá visualizar los valores de humedad del suelo obtenidos de los sensores instalados en el cultivo de fresas, con fecha y hora.	Alta	16
R07	Notificación de riego en el cultivo.	El usuario será notificado en la pantalla principal a través de un cuadro de mensaje cuando la humedad de suelo esté fuera del rango establecido.	Alta	13
Total				70

Tabla 35. Sprint Planning - Requerimientos para el Sprint 2.

A continuación, se presentan cada uno de los requerimientos con sus subtareas a realizarse en el Sprint 2, en conjunto con los story points de cada una.

Requerimiento	Subtareas	Story Points
R03 - Registro de información de cultivos.	1. Diseño e implementación de la interfaz para el registro de cultivos.	8
	2. Validar los campos en la interfaz de registro de cultivos.	3
	3. Guardar los datos en Cloud Firestore.	8
R04 - Visualización de información de los cultivos.	1. Diseño e implementación de la interfaz para la visualización de cultivos.	3
	2. Cargar los datos de Cloud Firestore en la aplicación web.	5
R05 - Edición de la información de los cultivos.	1. Diseño e implementación de la interfaz para la edición de cultivos.	3
	2. Implementación de la funcionalidad para habilitar y deshabilitar los campos para la edición de cultivos	3
	3. Guardar los datos en Cloud Firestore.	8

R06 - Visualización de los datos de los sensores.	1. Recuperar los datos de los sensores desde el servidor TTN mediante el protocolo de conexión MQTT usando el SDK de Node JS.	8
	2. Diseño e implementación de la interfaz para la visualización de los registros de los sensores.	8
R07 - Notificación de riego en el cultivo.	1. Diseño e implementación de la interfaz para la visualización de los registros de los sensores cuando se tengan valores bajos y altos de humedad.	5
	2. Diseño e implementación del mensaje a mostrarse cuando se tengan niveles bajos de humedad.	5
	3. Diseño e implementación del mensaje a mostrarse cuando se tengan niveles altos de humedad.	3
Total		70

Tabla 36. Tareas del Sprint 2.

4.4.4.2 Codificación

A continuación, se detalla la implementación de cada una de las tareas del Sprint 2.

Requisito R03: Registro de información de cultivos.

Cuando se ingresa por primera vez, en el módulo de cultivos se muestra la interfaz de la Figura 53, en la cual se tiene un botón que permite el registro del cultivo sobre el cual se va a registrar la información.



Figura 53. Interfaz del módulo de cultivos sin registros.

En la Figura 54 se muestra la interfaz de creación de cultivo con los campos Nombre del Cultivo, Tipo de Planta, Variedad, Cantidad de Plantas Sembradas, Extensión del cultivo (m2), Ubicación, Observaciones.



Figura 54. Requisito R03 – Pantalla de Registro de Cultivos.

De la misma manera, en la Figura 55 se muestra la validación de campos obligatorios en la interfaz de registro de cultivo.

Figura 55. Requisito R03 – Validación de campos.

En la Figura 56 se muestran los datos guardados en la base de datos de Cloud Firestore.

Actividad	Cantidad de Plantas	Especie de Cultivo	Ubicación	Código de Planta
SIEMPRE	10000	Cultivo de Planta	Toluca	REG12

Figura 56. Registro de cultivo en la base de datos de Cloud Firestore.

Requisito R04: Visualización de información de los cultivos.

En la Figura 57 se muestra la pantalla de visualización de la información registrada de cultivos.



The screenshot shows a web form titled "Cultivo de Fresas" with a sub-header "Editar cultivo" and a blue edit icon. The form contains several input fields: "Cultivo de Fresas", "Fresas", "Materia", "40000", "4000" (with a unit dropdown set to "m2"), "Tabla", and "Ninguna". All fields are currently disabled for editing.

Figura 57. Requisito R04 – Visualización de información de cultivos.

Requisito R05: Edición de la información de los cultivos.

En esta pantalla se tiene la opción de editar la información, en la cual se habilitan los campos para editar. Además aparecen los botones de Guardar y Cancelar como se muestra en la Figura 58.



This screenshot is identical to the previous one but shows the form in edit mode. The input fields are now active. At the bottom of the form, two buttons are visible: a grey "Cancelar" button and a blue "Guardar" button.

Figura 58. Requisito R05 – Edición de la información de cultivos.

Requisito R06: Visualización de los valores de los sensores.

Cuando se ingresa por primera vez y si el usuario no tiene sensores conectados en la aplicación se muestra la interfaz de la Figura 59.



Figura 59. Interfaz de la pantalla principal sin sensores.

Para poder obtener los datos de los sensores se utilizó el protocolo de conexión MQTT que usa el servidor TTN para el transporte de mensajes de publicación y suscripción. Se tienen bibliotecas cliente implementadas en diferentes lenguajes, en este caso se utilizó el SDK en Node JS [16].

Como primer paso se instaló en la aplicación el SDK con el siguiente comando: `npm install --save ttn.`

A continuación se debe importar el SDK desde el código del servidor de la aplicación web y agregar el ID de la aplicación (appID) y clave de acceso (accessKey) registradas en TTN. En la Figura 60 se pueden observar estos valores.

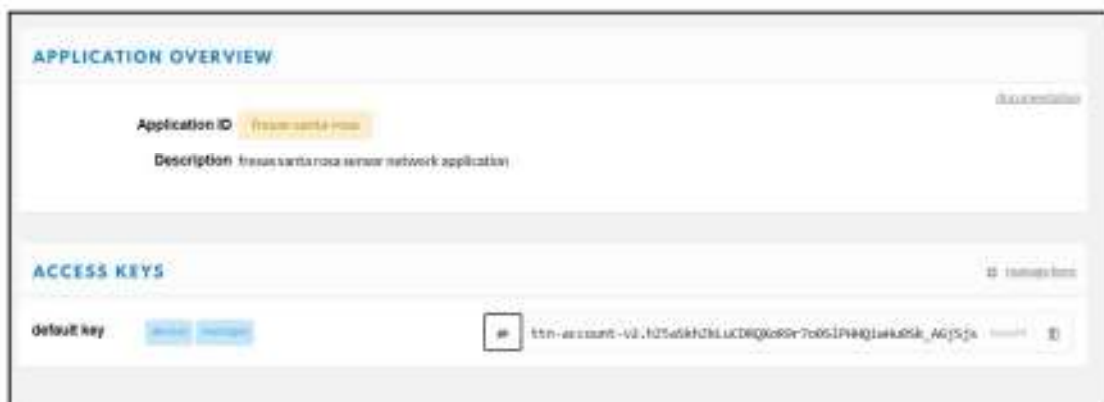


Figura 60. ID del dispositivo y clave de acceso de la aplicación registrados en TTN.

Finalmente se debe agregar un listener para recibir los datos desde los sensores. En la Figura 61 se muestran los datos recibidos en la consola del servidor de la aplicación web.

```
TERMINAL  PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN
C:\Users\MPika\Documents\Repositorios\proyecto\titulacion\test\app\code server.js
Connecting to the the things network data channel...
Connected to the things network data channel
Retrieving devices...
Retrieved devices (2)
Web server listening on port 3000!
retrieved uplink lsa001-01 { app_id: 'fresas-santa-rosa',
  dev_id: 'lsa01-01',
  hardware_serial: 'A849W1A92182486A',
  port: 2,
  counter: 3738,
  payload_raw: {buffer: [1c, 00, 00, 85, 50, 87, 2a, 1e, 14, 01]},
  payload_fields:
    { h20: 3.356,
      temp_DS18B20: '0.00',
      conduct_500L: '72.00',
      temp_500L: '18.14',
      water_500L: '13.00' },
  metadata:
    { time: '2021-01-27T00:43:45.422000Z',
      frequency: 900.1,
      modulation: 'QPSK',
      data_rate: '51.7Kb25',
      airtime: 61696000,
      coding_rate: '4/5',
      gateways: [ [Object] ] } }
```

Figura 61. Datos del sensor recibidos en la consola de la aplicación.

Estos datos son enviados al cliente de la aplicación web mediante la biblioteca socket.io y se muestran en la pantalla principal de la aplicación de la siguiente manera (Figura 62):

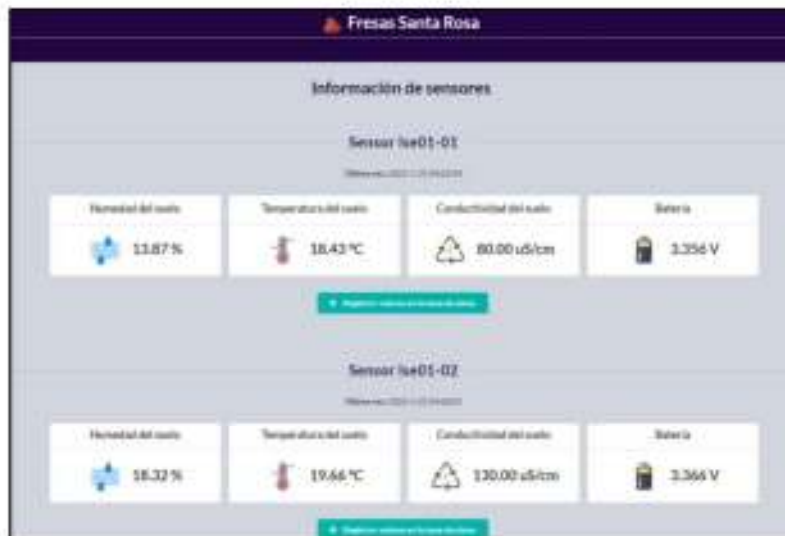


Figura 62. Requisito R06 - Visualización de los valores de los sensores.

En esta pantalla se muestran los siguientes datos de los sensores: Humedad, temperatura y conductividad del suelo, la batería y la fecha y hora de lectura del sensor. Adicional a

esto, también se tiene un botón por cada sensor para poder registrar los datos en Cloud Firestore. Cuando se selecciona este botón aparece un mensaje de aviso que indica que el registro fue creado en la base de datos como se muestra en la Figura 63.



Figura 63. Mensaje de creación de registro de los sensores en la base de datos.

En la Figura 64 se puede observar los datos registrados de los sensores en Cloud Firestore.

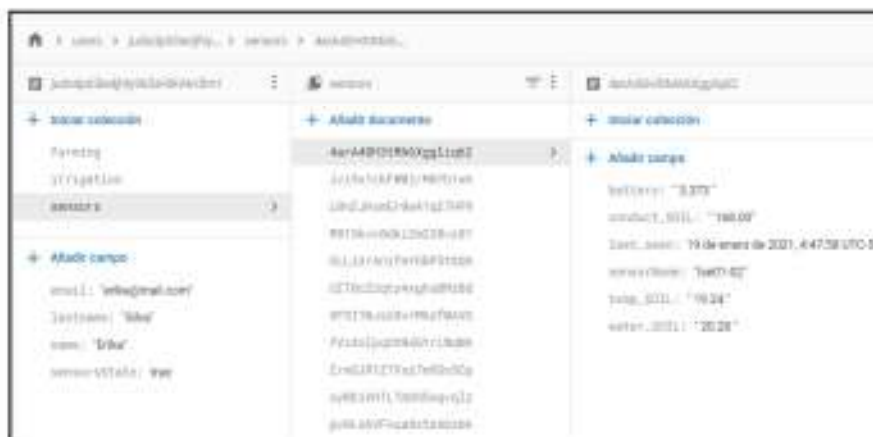


Figura 64. Información de los sensores registrados en Cloud Firestore

Requisito R07: Notificación de riego en el cultivo.

Para la implementación de este requisito, se estableció los niveles óptimos de humedad de suelo. Existen dos métodos utilizados para medir la humedad del suelo: el gravimétrico y volumétrico [50]. El método usado por los sensores Dragino es el volumétrico o VWC (*Volumetric Water Content*).

Según se menciona en [51]: "El contenido volumétrico de agua (VWC) es una medida de la cantidad de agua contenida en el suelo expresada como porcentaje de la mezcla total".

Esto indica que el volumen total del agua es solo un porcentaje del volumen total del suelo como se muestra en la Figura 65.

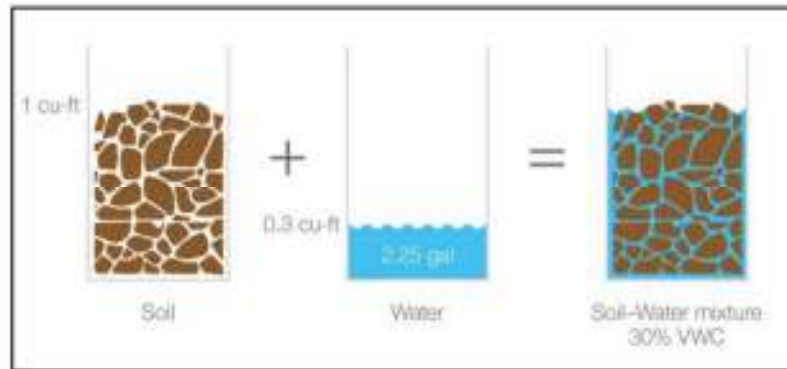


Figura 65. Contenido Volumétrico de Agua.

Una vez definido esto, en la Tabla 37 se muestran los niveles de humedad de suelo según [50], [51].

Niveles de Humedad del Suelo	Valor
Bajo	Menor a 10 %
Óptimo	Mayor a 10% y menor a 18%
Alto	Mayor a 18%

Tabla 37. Niveles de humedad del suelo.

En el caso de que se presentan niveles de humedad bajos se mostrará un mensaje en la parte superior de cada sensor indicando que se requiere realizar un riego en el cultivo. (Figura 66).



Figura 66. Mensaje de error para valores de humedad bajos.

Se muestra también el cuadro de los valores en color rojo, así como se aprecia en la Figura 67.



Figura 67. Requisito R07 – Notificación de riego en el cultivo.

4.4.4.3 Sprint Review

Una vez completada la implementación de las subtarefas de cada requisito del sprint 2, se presenta el producto funcional para obtener las observaciones y recomendaciones pertinentes. En la Tabla 38 se observan las observaciones encontradas.

Código	Nombre	Observaciones	Puntos estimados	Puntos finalizados	Puntos pendientes
R03	Registro de información de cultivos.	1. Colocar un mensaje de confirmación de registro de datos.	19	19	3
R04	Visualización de información de los cultivos.	Ninguna	8	8	0

R05	Edición de la información de los cultivos.	Ninguna	14	14	0
R06	Visualización de los valores de humedad del suelo.	Ninguna	16	16	0
R07	Notificación de riego en el cultivo.	<p>1. Diseñar e implementar la interfaz para la visualización de los registros de los sensores cuando se tengan valores altos de humedad.</p> <p>2. Diseñar e implementar el mensaje a mostrarse cuando se tengan niveles altos de humedad.</p>	13	7	6
Total			70	64	9

Tabla 38. Sprint Review del Sprint 2.

4.4.4.4 Despliegue

De igual manera, en el sprint 2 se realiza el despliegue de la aplicación mediante un servidor local usando los frameworks de Node JS y Express JS, para poder visualizar las funcionalidades implementadas. La ruta en la que se creó el servidor localmente fue <https://localhost:3000>.

4.4.5 Sprint 3

4.4.5.1 Sprint Planning

Código	Nombre	Descripción	Prioridad	Story Points
R03	Registro de información de cultivos.	[Procedencia: Sprint Review 2]. El usuario podrá confirmar el registro de datos a través de un mensaje en pantalla.	Alta	3
R07	Notificación de riego en el cultivo.	[Procedencia: Sprint Review 2]. El usuario podrá visualizar una notificación cuando los niveles de humedad estén por encima de 18%.	Alta	6
R08	Visualización de los sensores conectados en la aplicación.	El usuario podrá visualizar los sensores conectados a la aplicación web mediante su ID de dispositivo.	Media	5
R09	Buscar registros de los datos de sensores por fecha.	El usuario podrá buscar los registros de los valores de los sensores según la fecha indicada.	Media	13
R10	Eliminación de registros de datos de los sensores.	El usuario podrá eliminar los registros de valores de sensores menos relevantes.	Media	8
R11	Registro de información de riegos.	El usuario podrá ingresar la siguiente información de los riegos realizados en el cultivo de fresas: <ol style="list-style-type: none"> 1. Fecha 2. Hora 3. Duración 4. Humedad Actual 5. Observaciones 	Alta	19
Total				54

Tabla 39. Sprint Planning - Requerimientos para el Sprint 3.

A continuación, se presentan cada uno de los requerimientos con sus subtareas a realizarse en el Sprint 3, en conjunto con los story points de cada una.

Requerimiento	Subtareas	Story Points
[Procedencia: Sprint Review 2]. R03 - Registro de Información de cultivos.	1. Colocar un mensaje de confirmación de registro de datos.	3
[Procedencia: Sprint Review 2]. R07 - Notificación de riego en el cultivo.	1. Diseño e implementación de la interfaz para la visualización de los registros de los sensores cuando se tengan valores altos de humedad	3
	2. Diseño e implementación del mensaje a mostrarse cuando se tengan niveles altos de humedad.	3
R08 - Visualización de los sensores conectados en la aplicación.	1. Diseño e implementación de la interfaz para visualizar una lista de los sensores conectados a la aplicación mediante su ID de dispositivo.	5
R09 - Buscar registros de los datos de sensores por fecha.	1. Diseño e implementación de la interfaz para búsqueda de los datos de los sensores por la fecha de lectura del sensor.	5
	2. Recuperar los registros de la base de datos de Cloud Firestore que concuerden con la fecha de búsqueda.	3
	3. Mostrar los datos recuperados de la base de datos en la interfaz de búsqueda.	5
R10 - Eliminación de registros de datos de los sensores.	1. Agregar un botón con la acción de eliminar los registros de los datos de los sensores.	3
	2. Eliminar los registros de la colección de Sensores de la base de datos en Cloud Firestore	5
R11 - Registro de información de riegos.	1. Diseño e implementación de la interfaz para el registro de riegos.	8

	2. Validar los campos en la interfaz de registro de riegos.	3
	3. Guardar los datos en Cloud Firestore.	8
Total		54

Tabla 40. Tareas del Sprint 3.

4.4.5.2 Codificación

A continuación, se detalla la implementación de cada una de las tareas del Sprint 3.

Requisito R03: Registro de información de cultivos – Continuación del Sprint 2.

En la Figura 68 se muestra el mensaje de confirmación después de seleccionar el botón "Guardar" del formulario de registro de información de cultivos.



Figura 68. R03 – Registro de información de cultivos. Mensaje de confirmación de registro de datos.

Requisito R07: Notificación de riego en el cultivo – Continuación Sprint 2.

En la Figura 69 se muestra el mensaje cuando los valores de humedad están por encima del 18%. En este caso se muestra un mensaje en color azul.



Figura 69. R07 – Notificación de riego en el cultivo – Mensaje de información para valores altos de humedad.

Requisito R08: Visualización de los sensores conectados en la aplicación.

Cuando se ingresa por primera vez, en el módulo de Sensores se muestra la interfaz de la Figura 59, misma que presenta la pantalla principal cuando no hay sensores conectados a la aplicación.

En la Figura 70 se muestra la interfaz de un usuario que tiene sensores conectados a su aplicación. Los sensores se muestran en una lista con su ID del dispositivo y un botón que permite ingresar a otra pantalla que mostrará los datos del sensor registrados en la base de datos.

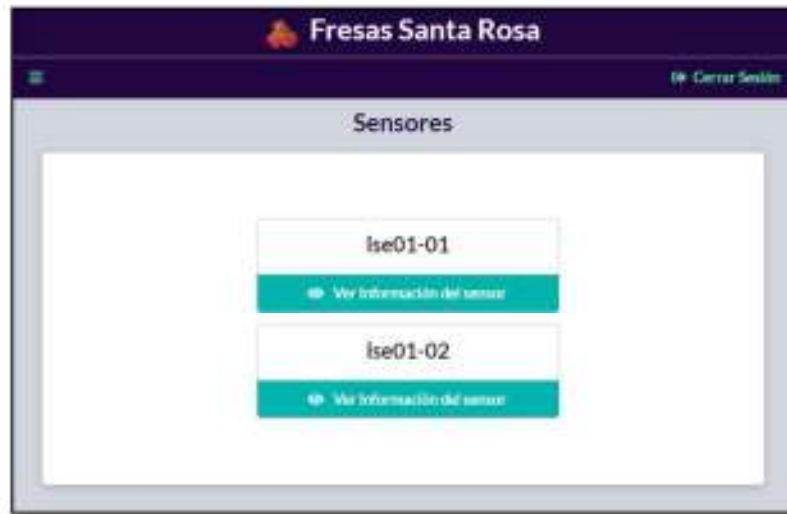


Figura 70. Requisito R08 – Pantalla de visualización de los sensores conectados en la aplicación.

Requisito R09: Buscar registros de los datos de sensores por fecha.

En la Figura 71 se muestra la pantalla de búsqueda de registros de los datos de los sensores que están registrados en la base de datos.

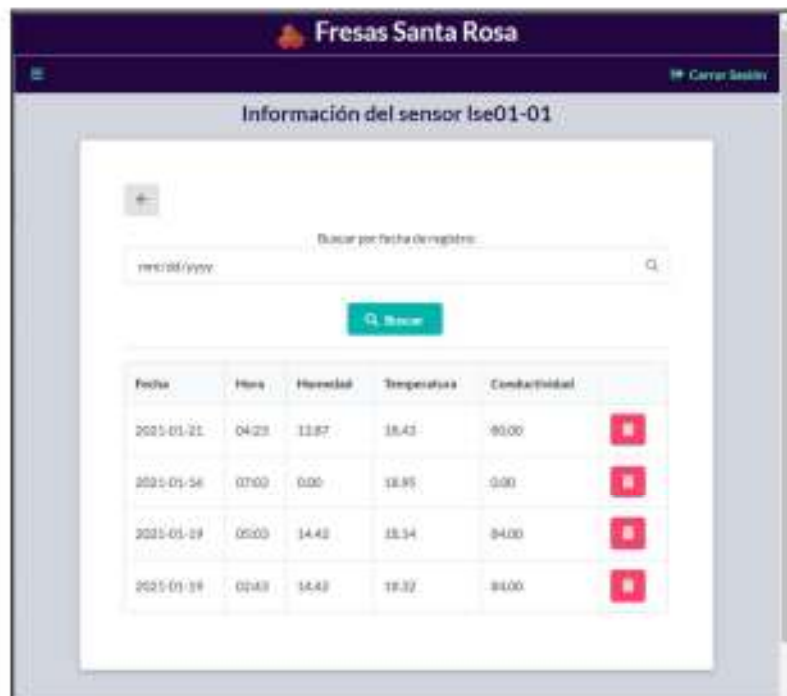


Figura 71. Requisito R09 – Pantalla de búsqueda de registros de datos de sensores.

En esta interfaz se permite la búsqueda de registros por fecha de lectura del sensor como se muestra en la Figura 72.



Figura 72. Ejemplo de búsqueda de registros de los sensores.

Adicional a esto, se tiene un botón de regresar que permite volver a la pantalla de la Figura 70 donde se muestran los sensores de la aplicación.



Figura 73. Botón regresar de la pantalla de búsqueda de los datos de sensores.

Requisito R10: Eliminación de registros de datos de los sensores.

En cada registro de la tabla que se muestra en la Figura 71 y Figura 72 se tiene un botón de eliminar como lo indica la Figura 74 que retira los registros de los datos de los sensores de la base de datos de Cloud Firestore.


Fecha	Hora	Humedad	Temperatura	Conductividad	Eliminar
2021-01-18	07:02	0.00	18.95	0.00	

Figura 74. Botón eliminar de la pantalla de búsqueda de los datos de sensores.

Requisito R11: Registro de información de riegos.

Cuando se ingresa por primera vez, en el módulo de Riegos se muestra la interfaz de la Figura 75.



Figura 75. Interfaz del módulo de Riegos.

La interfaz del módulo de Riego tiene los siguientes elementos:

1. Un botón para agregar nuevos registros de riego.
2. Un panel de búsqueda de registros por fecha de riego.
3. Una tabla que muestra los registros de riego.

Una vez que se selecciona el botón *Nuevo Registro de Riego* se abre la interfaz de la Figura 76.

Nuevo Riego

Fecha
02/04/2021

Hora
05:33 AM

Duración
15 minutos

Sensor
iso01-01

Humedad Actual
0%

Observaciones
Observaciones

Figura 76. Pantalla para agregar nuevo registro de riego.

En esta interfaz se tienen los siguientes campos:

1. Fecha de Riego
2. Hora de Riego
3. Duración (en minutos)
4. Humedad Actual del Terreno
5. Observaciones

De la misma manera, en la Figura 77 se muestra la validación de campos obligatorios en la interfaz de registro de riego.

Nuevo Riego

Fecha

 Campo obligatorio

Hora

 Campo obligatorio

Dirección

 Campo obligatorio

Humedad Actual del Terreno

 Campo obligatorio

Observaciones

Figura 77. Requisito R11 – Registro de información de riegos.

En la Figura 78 se muestran los datos guardados en la base de datos de Cloud Firestore.

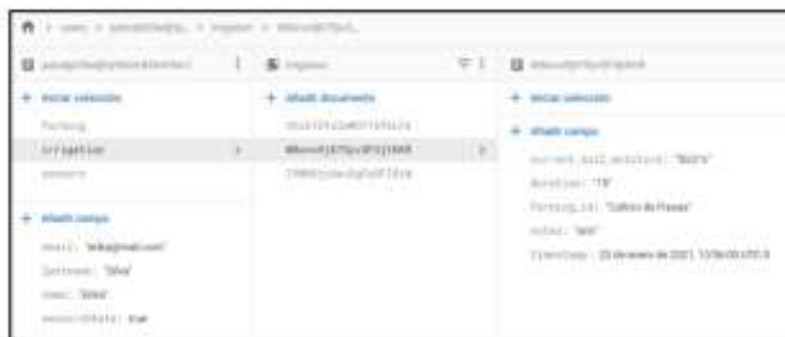


Figura 78. Registro de cultivo en la base de datos de Cloud Firestore.

4.4.5.3 Sprint Review

Una vez completada la implementación de las subtarefas de cada requisito del sprint 3, se presenta el producto funcional para obtener las observaciones y recomendaciones pertinentes. En la Tabla 41 se observan las observaciones encontradas.

Código	Nombre	Observaciones	Puntos estimados	Puntos finalizados	Puntos pendientes
R03	Registro de información de cultivos.	Ninguna	3	3	0
R07	Notificación de riego en el cultivo.	Ninguna	6	6	0
R08	Visualización de los sensores conectados en la aplicación.	Ninguna	5	5	0
R09	Buscar registros de los datos de sensores por fecha.	Ninguna	13	13	0
R10	Eliminación de registros de datos de los sensores.	1. Colocar un mensaje de confirmación antes de eliminar los registros.	8	8	3
R11	Registro de información de riegos.	Ninguna	19	19	0
Total			54	54	3

Tabla 41. Sprint Review del Sprint 3.

4.4.5.4 Despliegue

En el sprint 3 se realiza el despliegue de la aplicación mediante un servidor local usando los frameworks de Node JS y Express JS, para poder visualizar las funcionalidades implementadas. La ruta en la que se creó el servidor localmente fue <https://localhost:3000>.

4.4.6 Sprint 4

4.4.6.1 Sprint Planning

Código	Nombre	Descripción	Prioridad	Story Points
R10	Eliminación de registros de datos de los sensores.	El usuario podrá confirmar el la eliminación de registros a través de un mensaje en pantalla	Media	3
R12	Visualización de información de riegos.	El usuario podrá visualizar la información registrada de los riegos.	Alta	5
R13	Edición de información de riegos.	El usuario podrá editar los registros de riego ingresados en la aplicación web.	Media	22
R14	Eliminación de registros de riego.	El usuario podrá eliminar los registros ingresados en la aplicación web que sean irrelevantes.	Media	11
R15	Buscar registros de riego por fecha.	El usuario podrá buscar los registros de riego ingresados por la fecha.	Media	13
R16	Restablecimiento de contraseña.	El usuario podrá cambiar su contraseña para ingresar a la aplicación web.	Baja	5
R17	Visualización de la pantalla de inicio de la aplicación web.	El usuario podrá visualizar en la pantalla de inicio de la aplicación web una secuencia de imágenes del cultivo de fresas.	Media	8
Total				67

Tabla 42. Sprint Planning - Requerimientos para el Sprint 4.

A continuación, se presentan cada uno de los requerimientos con sus subtarear a realizarse en el Sprint 4, en conjunto con los story points de cada una.

Requerimiento	Subtarear	Story Points
---------------	-----------	--------------

R10 – Eliminación de registros de datos de los sensores.	1. Colocar un mensaje de confirmación antes de eliminar los registros.	3
R12 – Visualización de información de riegos.	1. Diseño e implementación de la interfaz para visualizar los registros de riego creados.	5
R13 – Edición de información de riegos.	1. Agregar un botón con la acción de editar los registros de riego.	3
	2. Diseño e implementación de la interfaz para la edición de los registros de riego.	3
	3. Implementación de la funcionalidad para cargar los datos en la interfaz de edición de riegos según el elemento seleccionado de la tabla.	8
	3. Guardar los datos en Cloud Firestore.	8
R14 – Eliminación de registros de riego.	1. Agregar un botón con la acción de eliminar los registros de riego.	3
	2. Colocar un mensaje de confirmación antes de eliminar los registros.	3
	3. Eliminar los registros de la colección de Riego de la base de datos en Cloud Firestore	5
R15 – Buscar registros de riego por fecha.	1. Diseño e implementación de la interfaz para búsqueda de los registros de riego del cultivo.	5
	2. Recuperar los registros de la base de datos de Cloud Firestore que concuerden con la fecha de búsqueda.	3
	3. Mostrar los datos recuperados de la base de datos en la interfaz de búsqueda.	5
R16 – Restablecimiento de contraseña.	1. Diseño e implementación de una interfaz que permita realizar el restablecimiento de contraseña en el caso de que un usuario haya olvidado su contraseña.	5
R17 – Visualización de la pantalla de inicio de la aplicación web.	1. Diseño e implementación de una interfaz de inicio donde se presente una secuencia de imágenes del cultivo de fresas.	8
Total		67

Tabla 43. Tareas del Sprint 4.

4.4.6.2 Codificación

A continuación, se detalla la implementación de cada una de las tareas del Sprint 4.

Requisito R10: Eliminación de registros de datos de los sensores – Continuación Sprint 3.

En la Figura 79 se muestra el mensaje de confirmación para eliminar registros después de seleccionar el botón eliminar en la tabla.



Figura 79. R10 – Eliminación de registros de datos de los sensores. Mensaje de confirmación de eliminación de registros.

Requisito R12: Visualización de información de riegos.

En la Figura 80 se muestra la interfaz principal del módulo de Riegos con un registro cargado desde la base de datos.

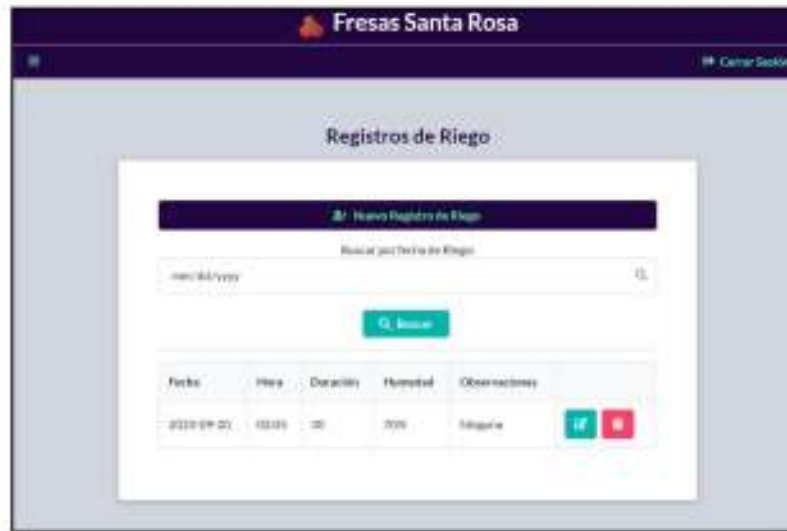


Figura 80. Requisito R12 – Visualización de información de riegos.

Requisito R13: Edición de información de riegos.

En cada registro de la tabla de riegos se tiene un botón de editar como se aprecia en la Figura 85. Este botón abre la interfaz de edición con los datos del registro seleccionado de la tabla.







Fecha	Hora	Duración	Humedad	Observaciones	Editar
2020-09-20	02:05	30	70%	Ninguna	 
2021-01-23	10:56	15	30.0%	nro	 
2021-01-23	02:55	15	0.0%		 

Figura 81. Tabla de registros de riego con el botón editar.

En la Figura 82 se muestra la interfaz de edición con un registro seleccionado.



Figura 82. Requisito R13 – Edición de información de riegos.

Requisito R14: Eliminación de registros de riego.

En cada registro de la tabla que se muestra en la Figura 81 se tiene un botón de eliminar (Figura 83) que elimina los registros de riego de la base de datos de Cloud Firestore.





Fecha	Hora	Duración	Humedad	Observaciones	Eliminar
2021-01-23	02:55	15	0,0%		 
2021-01-23	10:56	15	50,0%	nm	 

Figura 83. Botón eliminar de la pantalla de búsqueda de los registro de riego.

Antes de eliminar los registros de la base de datos se presenta un mensaje de confirmación al usuario (Figura 84).



Figura 84. Mensaje de confirmación antes de eliminar registros de riego.

Requisito R15: Buscar registros de riego por fecha.

En la Figura 85 se muestra la pantalla de búsqueda de registros de riego que fueron registrados en la base de datos.

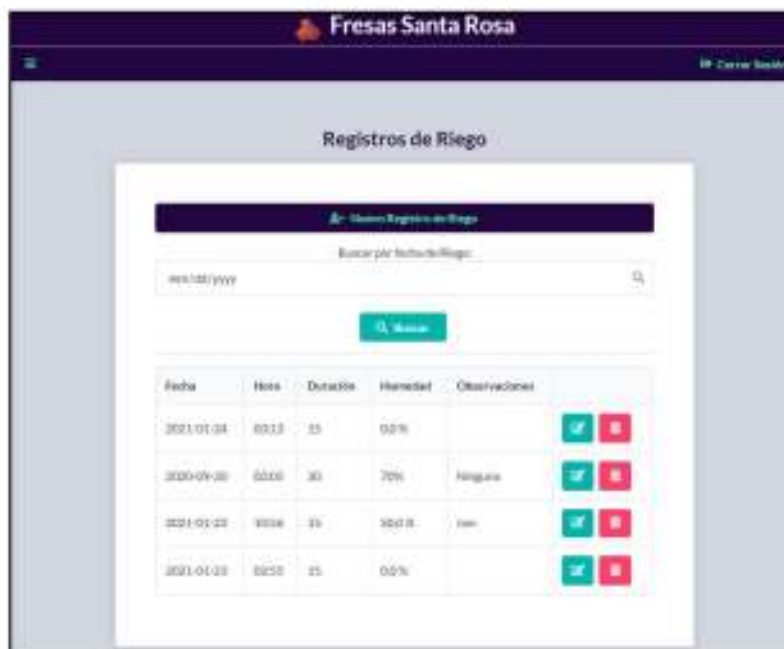


Figura 85. Requisito R09 – Pantalla de búsqueda de registros de datos de sensores.

En esta interfaz se permite la búsqueda por fecha de registro como se muestra en la Figura 86.



Figura 86. Ejemplo de búsqueda de registros de los sensores.

Requisito R16: Restablecimiento de contraseña.

En la Figura 87 se muestra la interfaz para realizar el restablecimiento de contraseña. El único campo para ingresar es el correo electrónico mediante el cual Firebase Authentication enviará un email con el enlace para el cambio de contraseña.



Figura 87. Requisito R16 – Restablecimiento de contraseña

Al seleccionar el botón *Recuperar Contraseña*, Firebase Authentication envía un email con el enlace para el cambio de contraseña. En la Figura 88 se puede observar el email enviado.



Figura 88. Email de restablecimiento de contraseña.

Este enlace abre una página en donde se muestra una interfaz que permite el ingreso de una nueva contraseña (Figura 89).



Figura 89. Interfaz de Cambio de Contraseña de Firebase Authentication.

Firebase Authentication tiene la opción de editar el contenido del correo electrónico que será enviado. En la Figura 90 se muestra la plantilla del correo electrónico de cambio de contraseña.

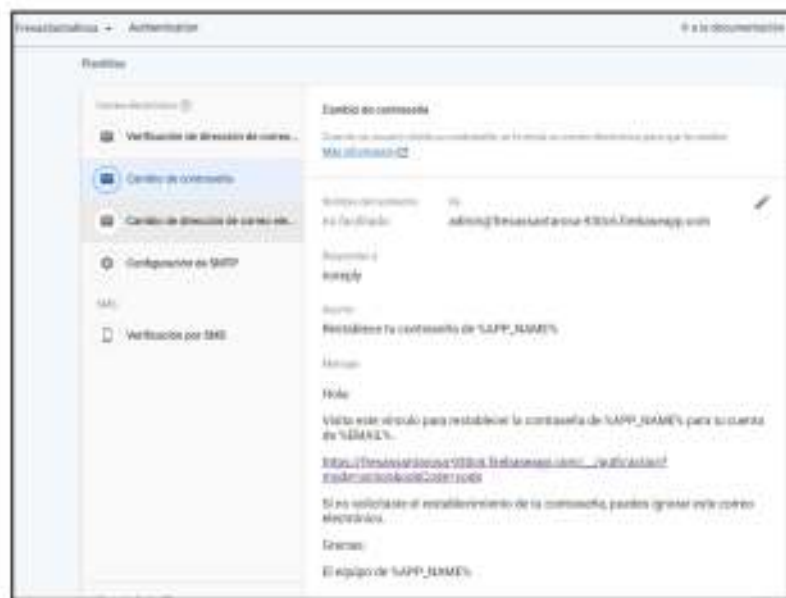


Figura 90. Plantilla del correo electrónico para el cambio de contraseña de Firebase Authentication.

Requisito R17: Visualización de la pantalla de inicio de la aplicación web.

En la pantalla de inicio de la aplicación web, además de los botones de *Inicio de Sesión* y *Registrarse*, se tiene una secuencia de imágenes relacionadas con el cultivo de fresas como se aprecia en las Figura 91 y Figura 92.



Figura 91. Requisito R17 – Visualización de la pantalla de inicio de la aplicación web.



Figura 92. Ejemplo de Visualización de la secuencia de imágenes de la pantalla de inicio.

4.4.6.3 Sprint Review

Una vez completada la implementación de las subtarefas de cada requisito del sprint 4, se presenta el producto funcional para obtener las observaciones y recomendaciones pertinentes. En la se observan las observaciones encontradas.

Código	Nombre	Observaciones	Puntos estimados	Puntos finalizados	Puntos pendientes
R10	Eliminación de registros de datos de los sensores.	Ninguna	3	3	0
R12	Visualización de información de riegos.	Ninguna	5	5	0
R13	Edición de información de riegos.	Ninguna	22	22	0
R14	Eliminación de registros de riego.	Ninguna	11	11	0
R15	Buscar registros de riego por fecha.	Ninguna	13	13	0
R16	Restablecimiento de contraseña.	Ninguna	5	5	0
R17	Visualización de la pantalla de inicio de la aplicación web.	Ninguna	8	8	0
Total			67	67	0

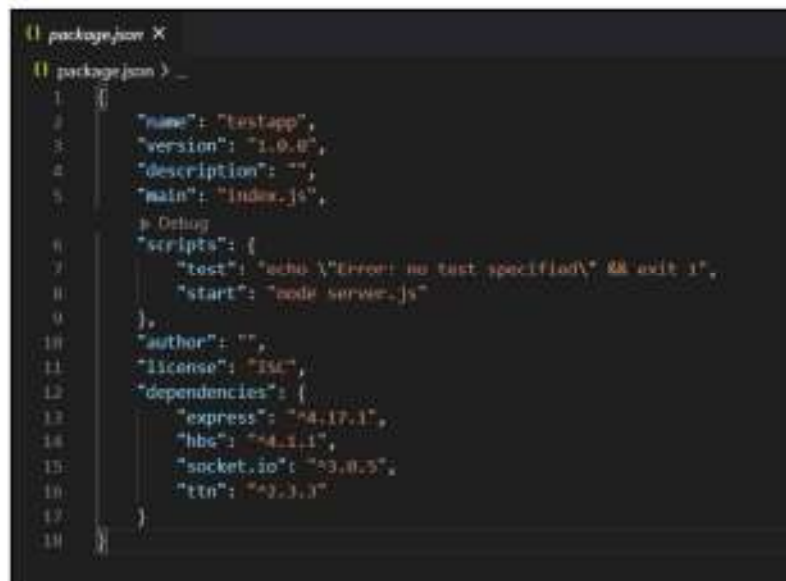
Tabla 44. Sprint Review del Sprint 4.

4.4.6.4 Despliegue

Una vez finalizada la implementación de los requerimientos de la aplicación web se realizó el despliegue de esta. Para ello se utilizó el servicio de Heroku que provee los componentes necesarios para administrar una aplicación web de manera sencilla.

Para realizar el despliegue de la aplicación web con el framework de Node.js se debe configurar los siguientes archivos:

1. **package.json**: contiene las características para iniciar la aplicación. Uno de los elementos que se debe configurar correctamente es el script de start que debe ser el mismo del archivo *Procfile*. En la Figura 93 se muestra el contenido del archivo *package.json*.



```
package.json X
package.json > _
1  {}
2    "name": "testapp",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \\\"Error: no test specified\\\" && exit 1",
8      "start": "node server.js"
9    },
10   "author": "",
11   "license": "ISC",
12   "dependencies": {
13     "express": "^4.17.1",
14     "hbs": "^4.1.1",
15     "socket.io": "^3.0.5",
16     "ttn": "^3.1.3"
17   }
18 }
```

Figura 93. Contenido del archivo *package.json* para el despliegue de la aplicación.

2. **Procfile**: contiene el nombre del archivo para iniciar el servidor (Figura 94).



```
Procfile X
Procfile
1  echo "web:node:server.js" > Procfile
```

Figura 94. Contenido del archivo *Procfile* para el despliegue de la aplicación.

Esta plataforma utiliza Git para el transporte del código fuente hacia Heroku. Una vez creada la aplicación en este servicio, se asocia a un control remoto de Git llamado Heroku.

Por lo cual la única línea de código para desplegar nuevas funcionalidades es: `git push heroku master`.

Para realizar por primera vez el despliegue de la aplicación se ejecutaron los siguientes comandos:

1. `git init`
2. `heroku login -i`
3. `heroku create fresas-app`
4. `git add .`
5. `git commit -m "Ready to deploy"`
6. `git push heroku master`

En la Figura 95 se puede apreciar la aplicación web desplegada en Heroku en la dirección URL <https://fresas-app.herokuapp.com/>.

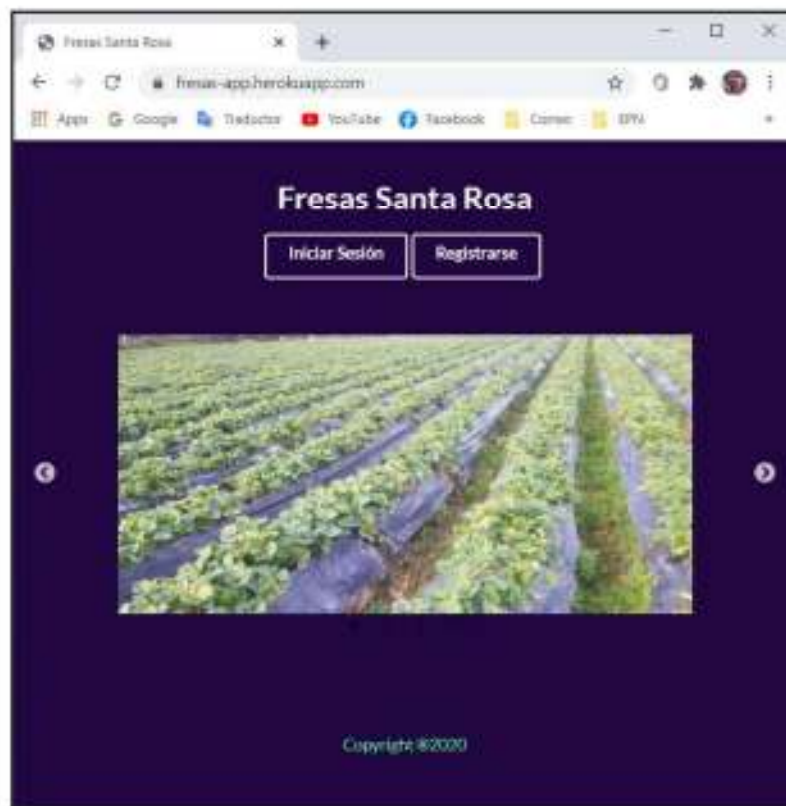


Figura 95. Aplicación web desplegada en Heroku.

En el Anexo 7: Video demostrativo de la funcionalidad de la Aplicación Web se muestra un video con la funcionalidad completa del prototipo implementado.

CAPÍTULO 5. RESULTADOS Y DISCUSIÓN

5.1 Pruebas de funcionalidad

La funcionalidad de la aplicación web se probó por la parte interesada y la desarrolladora en cada uno de los sprint implementados. En esta parte, se realizaron las pruebas de funcionalidad de la aplicación con las personas relacionadas al negocio y personas externas. Para llevar a cabo estas pruebas, se utilizó el formato de caso de prueba que se muestra en la Tabla 45.

Caso de prueba – N.º			
Funcionalidad			
Descripción			
Entradas			
Proceso			
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
Observaciones			

Tabla 45. Formato de caso de prueba para pruebas de funcionalidad.

Se realizó 8 casos de prueba para las funcionalidades más relevantes de la aplicación y fue ejecutado por 8 personas entre miembros del negocio y personas externas. En el

Anexo 2: Resultados de los Casos de Prueba se puede observar a mayor detalle los resultados de cada caso de prueba realizado. A continuación, se detalla las plantillas de cada caso de prueba

5.1.1 Casos de prueba

La Tabla 46 muestra la plantilla para el caso de prueba N°1 de creación de cuenta e inicio de sesión.

Caso de prueba – N°1	
Funcionalidad	Creación de cuenta e inicio de sesión.
Descripción	El usuario realizará la creación de una cuenta e iniciará sesión en la aplicación web
Entradas	<i>Datos para creación de cuenta</i>

	Nombres:		
	Apellidos:		
	Correo electrónico:		
	Contraseña:		
Proceso	1. Acceder a la dirección URL https://fresas-app.herokuapp.com/ .		
	2. Hacer click en el botón "Registrarse".		
	3. Ingresar los datos solicitados.		
	4. Hacer click en el botón "Crear Cuenta".		
	5. En la interfaz de inicio de sesión, ingresar los datos solicitados (correo electrónico y contraseña).		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo crear una cuenta en la aplicación web.			
El usuario pudo acceder a la aplicación web.			
Observaciones			

Tabla 46. Caso de Prueba N°1 – Creación de una cuenta e inicio de sesión.

La Tabla 47 muestra la plantilla para el caso de prueba N°2 de registro de información del cultivo.

Caso de prueba – N°2		
Funcionalidad	Registro de cultivo.	
Descripción	El usuario realizará el registro de la información del cultivo.	
Entradas	<i>Datos para el registro de cultivo.</i>	
	Nombre del cultivo:	
	Tipo de planta:	
	Variedad:	
	Cantidad de Plantas Sembradas:	
	Extensión del Cultivo:	
	Ubicación:	
	Observaciones:	

Proceso	1. Una vez iniciado sesión en la aplicación web, hacer click en el menú de navegación y seleccionar la opción "Cultivos".		
	2. En la interfaz mostrada, hacer click en el botón "Registrar un Cultivo".		
	3. En el formulario de información de cultivo llenar los siguientes campos: 1. Nombre del cultivo 2. Tipo de planta 3. Variedad 4. Cantidad de Plantas Sembradas 5. Extensión del Cultivo 6. Ubicación 7. Observaciones		
	4. En el final del formulario, hacer click en el botón "Guardar".		
	5. Un mensaje de éxito será mostrado cuando se haya creado el registro correctamente.		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo registrar la información de un cultivo.			
Observaciones			

Tabla 47. Caso de Prueba N°2 – Registro de cultivo.

La Tabla 48 muestra la plantilla para el caso de prueba N°3 de registro de un riego.

Caso de prueba – N°3	
Funcionalidad	Registro de riego.
Descripción	El usuario realizará el registro de un riego.
Entradas	<i>Datos para el registro de riego.</i>
	Fecha:
	Hora:
	Duración:
	Humedad actual del terreno:
	Observaciones:
Proceso	1. Una vez iniciado sesión en la aplicación web, hacer click en el menú de navegación y seleccionar la opción "Riegos".
	2. En la interfaz mostrada, hacer click en el botón "Nuevo Registro de Riego".
	3. En el formulario de información de riego llenar los siguientes campos: 1. Fecha 2. Hora 3. Duración

	4. Humedad actual del terreno 5. Observaciones		
	4. En el final del formulario, hacer click en el botón "Guardar".		
	5. Un mensaje de éxito será mostrado cuando se haya creado el registro correctamente.		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo crear un nuevo registro de riego.			
Observaciones			

Tabla 48. Caso de Prueba N°3 – Registro de riego.

La Tabla 49 muestra la plantilla para el caso de prueba N°4 de búsqueda de registros de valores de los sensores en una fecha específica.

Caso de prueba – N°4			
Funcionalidad	Búsqueda de registros de los valores de los sensores por fecha.		
Descripción	El usuario realizará la búsqueda de registros de valores de los sensores en una fecha especificada.		
Entradas	<i>Datos para la búsqueda de registros.</i>		
	Fecha de registro:		
Proceso	1. Una vez iniciado sesión en la aplicación web, hacer click en el menú de navegación y seleccionar la opción "Sensores".		
	2. En la interfaz mostrada, seleccionar un sensor y hacer click en el botón "Ver Información del sensor".		
	3. En la interfaz mostrada ingresar la fecha de registros a buscar.		
	4. Hacer click en el botón "Buscar".		
	5. La tabla de registros mostrará los registros en la fecha especificada.		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo realizar la búsqueda de registros de valores de los sensores según una fecha específica.			
Observaciones			

Tabla 49. Caso de Prueba N°4 – Búsqueda de registros de los valores de los sensores por fecha.

La Tabla 50 muestra la plantilla para el caso de prueba N°5 de edición de un registro seleccionado de la tabla de registros de Riegos.

Caso de prueba – N°5			
Funcionalidad	Editar un registro de riego.		
Descripción	El usuario realizará la edición de un registro de riego seleccionado de la tabla de registros guardados en la base de datos.		
Entradas	<i>Datos para la edición de registro de riego.</i>		
	Fecha:		
	Hora:		
	Duración:		
	Humedad actual del terreno:		
Observaciones:			
Proceso	1. Una vez iniciado sesión en la aplicación web, hacer click en el menú de navegación y seleccionar la opción "Riegos".		
	2. En la interfaz mostrada, seleccionar un registro de riego y hacer click en el botón "Editar".		
	3. En el formulario de edición de riego llenar los campos que crea convenientes:		
	<ol style="list-style-type: none"> 1. Fecha 2. Hora 3. Duración 4. Humedad actual del terreno 5. Observaciones 		
	4. Hacer click en el botón "Guardar".		
5. La tabla de registros se actualizará con los cambios realizados.			
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo realizar la edición de registros de riegos.			
Observaciones			

Tabla 50. Caso de Prueba N°5 – Editar un registro de riego.

La Tabla 51 muestra la plantilla para el caso de prueba N°6 de eliminación de un registro de la tabla de registros de valores de los sensores.

Caso de prueba – N°6	
Funcionalidad	Eliminación de un registro de valores de los sensores.
Descripción	El usuario realizará la eliminación de un registro de valores de los sensores.
Entradas	Ninguna

Proceso	1. Una vez iniciado sesión en la aplicación web, hacer click en el menú de navegación y seleccionar la opción "Sensores".		
	2. En la interfaz mostrada, seleccionar un sensor y hacer click en el botón "Ver Información del Sensor".		
	3. En la interfaz mostrada, seleccionar un registro de riego y hacer click en el botón "Eliminar".		
	4. Confirmar la eliminación del registro.		
	5. La tabla de registros se actualizará con los cambios realizados.		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo realizar la eliminación de registros de valores de los sensores.			
Observaciones			

Tabla 51. Caso de Prueba N°6 – Eliminar un registro de valores de los sensores.

La Tabla 52 muestra la plantilla para el caso de prueba N°7 de visualización de valores de los sensores del cultivo.

Caso de prueba – N°7			
Funcionalidad	Visualización de valores de los sensores del cultivo.		
Descripción	El usuario visualizará en la pantalla principal, los valores obtenidos de los sensores.		
Entradas	<i>Datos para inicio de sesión</i>		
	Correo electrónico:		
	Contraseña:		
Proceso	1. Acceder a la dirección URL https://fresas-app.herokuapp.com/ .		
	2. Hacer click en el botón "Inicio de sesión".		
	3. Ingresar los datos solicitados. <ul style="list-style-type: none"> 1. Correo electrónico 2. Contraseña 		
	4. Hacer click en el botón "Iniciar Sesión".		
	5. En la interfaz mostrada, puede observar los datos obtenidos de los sensores por cada uno de ellos.		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo visualizar los datos obtenidos de los sensores en la pantalla de inicio.			
Observaciones			

Tabla 52. Caso de Prueba N°7 – Visualización de valores de los sensores del cultivo.

La Tabla 53 muestra la plantilla para el caso de prueba N°8 de visualización de mensajes de advertencia según los niveles de humedad del suelo establecidos.

Caso de prueba – N°8			
Funcionalidad	Visualización de mensajes de advertencia según los niveles de humedad del suelo.		
Descripción	El usuario visualizará en la pantalla principal, los valores obtenidos de los sensores y mensajes de advertencia según los niveles de humedad del suelo establecidos.		
Entradas	<i>Datos para inicio de sesión</i>		
	Correo electrónico:		
	Contraseña:		
Proceso	1. Acceder a la dirección URL https://fresas-app.herokuapp.com/ .		
	2. Hacer click en el botón "Inicio de sesión".		
	3. Ingresar los datos solicitados. <ol style="list-style-type: none"> 1. Correo electrónico 2. Contraseña 		
	4. Hacer click en el botón "Iniciar Sesión".		
	5. En la interfaz mostrada, puede observar los datos obtenidos de los sensores por cada uno de ellos.		
	6. Cada 20 minutos observará cambios en los datos. Si la humedad está por debajo de 10%, se mostrará un mensaje indicando la falta de agua en el cultivo. Por el contrario, si la humedad está por encima de 18% se mostrará un mensaje indicando que la humedad es demasiado alta.		
Resultado Esperado	Resultado Obtenido		
	Si	Parcialmente	No
El usuario pudo visualizar los mensajes de advertencia cuando la humedad está fuera de los rangos establecidos.			
Observaciones			

Tabla 53. Caso de Prueba N°8 – Visualización de mensajes de advertencia cuando la humedad está fuera del rango establecido.

5.1.2 Resultados Globales

Después de haber realizado los casos de prueba se realizó una encuesta (Anexo 3: Encuesta de Pruebas de Funcionalidad) a las 8 personas que realizaron los casos de prueba entre los rangos de edad de 18 a 50 años, para determinar que se hayan cumplido los resultados esperados.

En la Figura 96 se puede observar los rangos de edad de las personas que realizaron la encuesta.

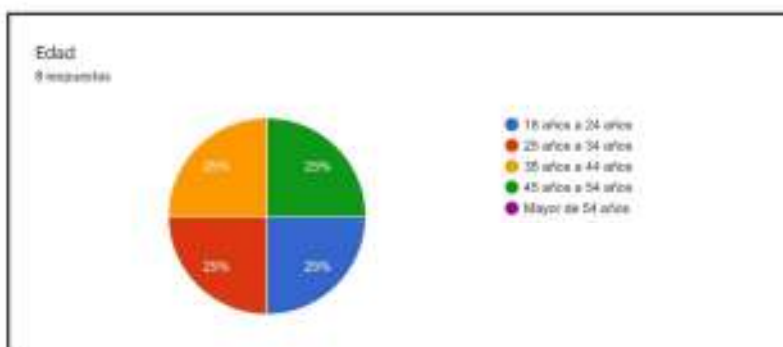


Figura 96. Distribución de edades en las pruebas de la aplicación.

En la Figura 97 se puede observar el campo de trabajo al que pertenecen las personas que fueron encuestadas.

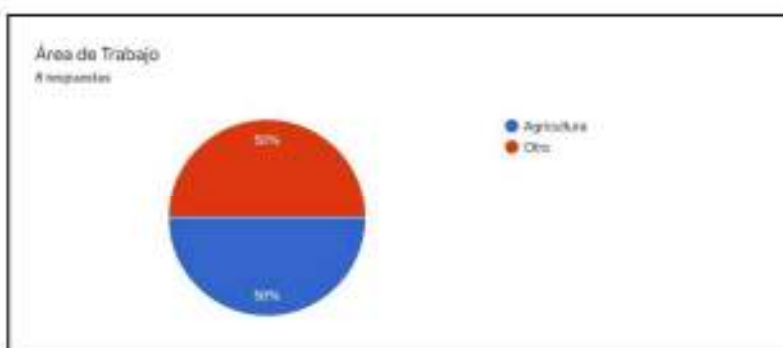


Figura 97. Distribución del área de trabajo en las pruebas de la aplicación.

De la misma manera, en la Figura 98 se muestra los resultados por cada pregunta realizada a los 8 participantes. En la mayoría de las preguntas se obtuvo un resultado satisfactorio, pero como se puede apreciar en la última pregunta que verifica si la aplicación muestra notificaciones según los niveles de humedad del suelo, se obtuvo una respuesta parcial.

Este resultado puede ser debido a que la aplicación muestra mensajes según los niveles que se obtengan de los sensores y en el momento que los usuarios se encontraban en la aplicación no obtuvieron ningún mensaje u obtuvieron al menos uno de los mensajes. La razón principal es que el cultivo estaba en los niveles óptimos de humedad y por ende la aplicación no mostró mensajes.



Figura 98. Gráfico de resultados de la encuesta de pruebas de funcionalidad.

Por último, en la Figura 99 se muestran los resultados globales obtenidos en las pruebas de funcionalidad. Como se puede apreciar se tiene un 94% de usuarios que obtuvieron los resultados esperados en cada caso de prueba.



Figura 99. Resultados globales de la encuesta de funcionalidad.

En el Anexo 5: Resultados de las Encuestas de Pruebas de Funcionalidad se puede observar a mayor detalle los resultados de las encuestas realizadas.

5.2 Pruebas de Usabilidad

Las pruebas de usabilidad indican la facilidad de uso de la aplicación desde el punto de vista del usuario. Para realizar estas pruebas se utilizó la herramienta SUS (*System Usability Scale*) que consiste en un cuestionario de 10 preguntas con 5 opciones de

respuesta en una escala del 1 al 5 donde 1 corresponde a *Totalmente de acuerdo* y 5 corresponde a *Totalmente en desacuerdo* [52].

Las preguntas del cuestionario según se menciona en [52] se encuentran listadas a continuación:

- ¿Cree que usaría esta aplicación frecuentemente?
- ¿Encuentra esta aplicación innecesariamente compleja?
- ¿Cree que la aplicación es fácil de usar?
- ¿Cree que necesitaría apoyo técnico para utilizar la aplicación?
- ¿Encontró diversas funciones de la aplicación que estaban bien integradas?
- ¿Piensa que hay demasiada inconsistencia en la aplicación?
- ¿Piensa que la mayoría de la gente aprendería a utilizar este sistema de manera rápida?
- ¿Encontró la aplicación muy complicada de usar?
- ¿Se sintió muy seguro al usar la aplicación?
- ¿Necesitó aprender varias cosas antes de poder utilizar la aplicación?

Estas preguntas fueron contestadas después de ejecutar cada caso de prueba.

5.2.1 Resultados individuales

En esta sección se muestran los resultados de cada una de las preguntas listadas anteriormente usando la herramienta SUS. Todas las preguntas tienen una escala del 1 (Totalmente en desacuerdo) al 5 (Totalmente de acuerdo).

Resultados de la Pregunta N°1: ¿Cree que usaría esta aplicación frecuentemente?

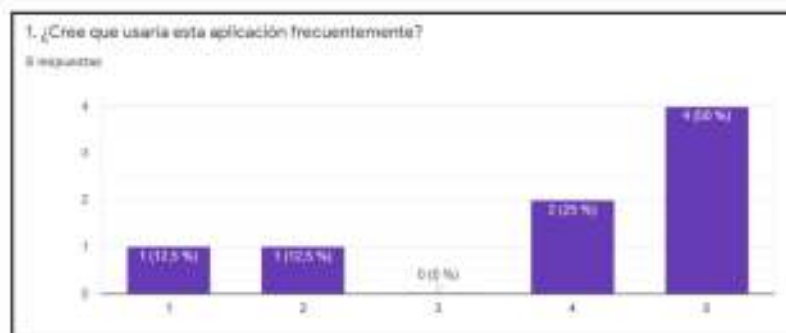


Figura 100. Resultados de la Pregunta N°1 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°2: ¿Encuentra esta aplicación innecesariamente compleja?

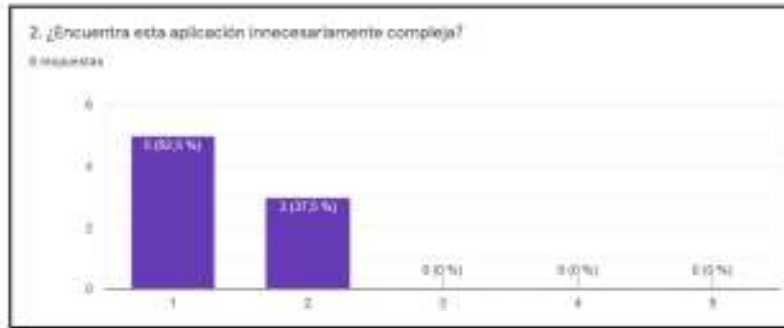


Figura 101. Resultados de la Pregunta N°2 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°3: ¿Cree que la aplicación es fácil de usar?

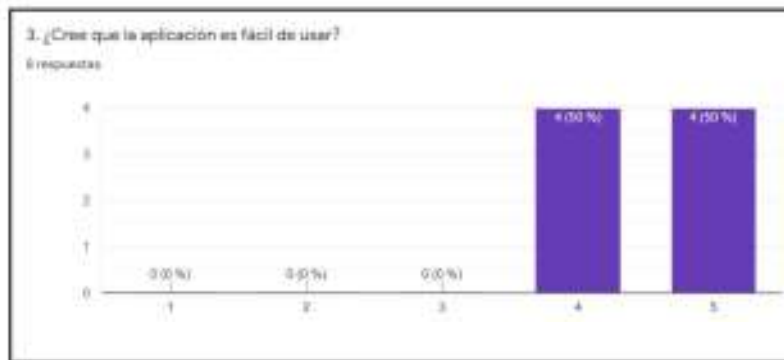


Figura 102. Resultados de la Pregunta N°3 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°4: ¿Cree que necesitaría apoyo técnico para utilizar la aplicación?

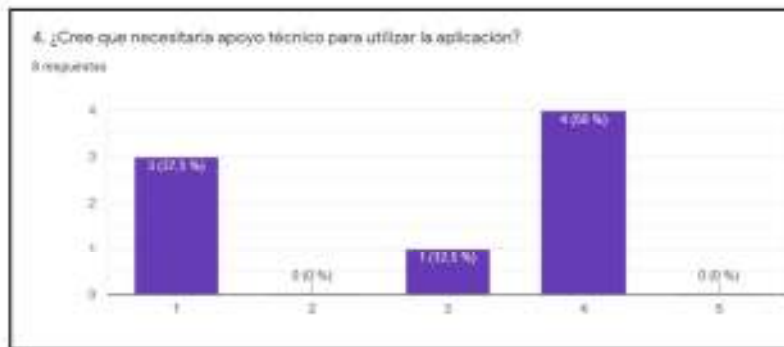


Figura 103. Resultados de la Pregunta N°4 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°5: ¿Encontró diversas funciones de la aplicación que estaban bien integradas?



Figura 104. Resultados de la Pregunta N°5 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°6: ¿Piensa que hay demasiada inconsistencia en la aplicación?



Figura 105. Resultados de la Pregunta N°6 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°7: ¿Piensa que la mayoría de la gente aprendería a utilizar este sistema de manera rápida?

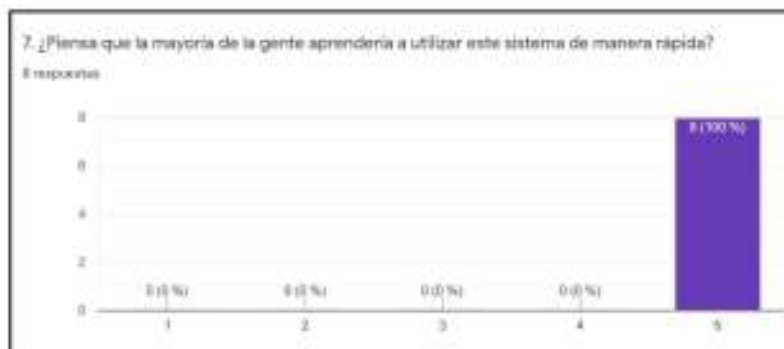


Figura 106. Resultados de la Pregunta N°7 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°8: ¿Encontró la aplicación muy complicada de usar?

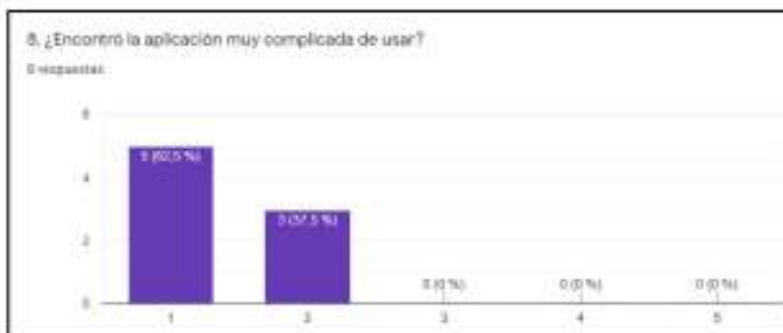


Figura 107. Resultados de la Pregunta N°8 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°9: ¿Se sintió muy seguro al usar la aplicación?

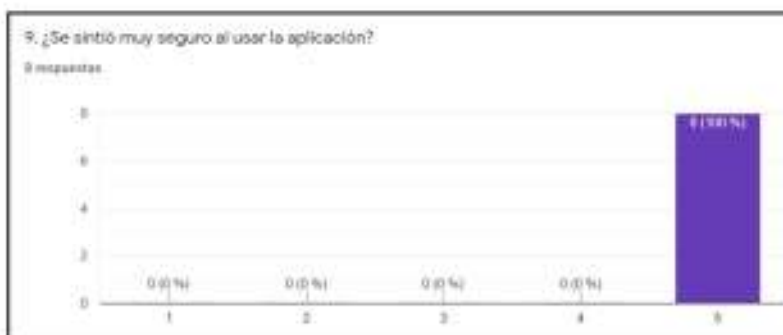


Figura 108. Resultados de la Pregunta N°9 de la Encuesta de Usabilidad.

Resultados de la Pregunta N°10: ¿Necesitó aprender varias cosas antes de poder utilizar la aplicación?

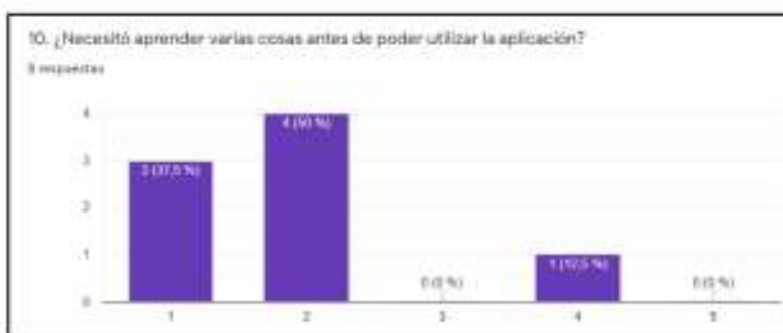


Figura 109. Resultados de la Pregunta N°10 de la Encuesta de Usabilidad.

5.2.2 Resultados Globales

En esta sección se muestra los resultados globales de las preguntas descritas anteriormente después de haber normalizado estos resultados según la escala de calificación de SUS.

En la Figura 110 se muestra un gráfico que indica el promedio de respuesta obtenido por cada pregunta después de haber realizado la normalización.

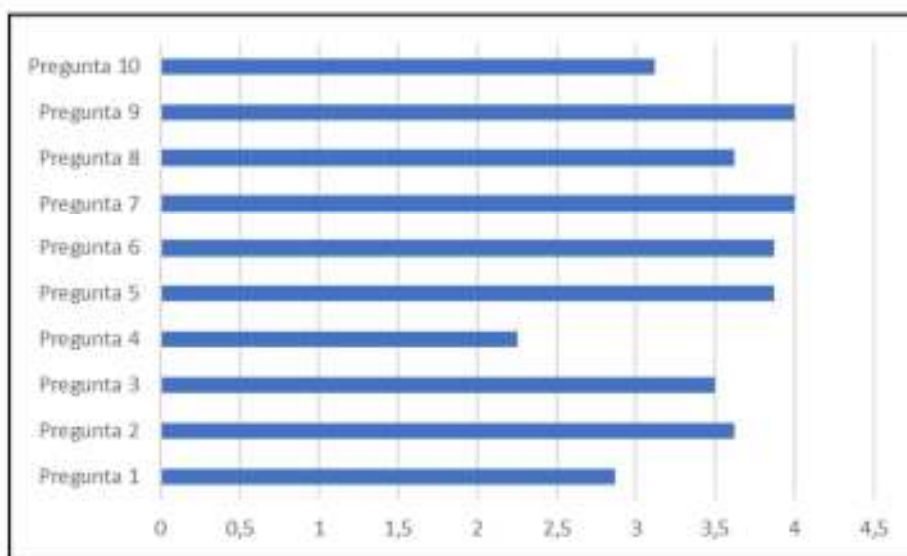


Figura 110. Gráfico de los puntajes promedio para cada pregunta de la encuesta de usabilidad según la escala SUS.

En general, se puede apreciar que las preguntas con puntaje más bajo son la pregunta 1, 4 y 10. La pregunta 4 (¿Cree que necesitaría apoyo técnico para utilizar la aplicación?) obtuvo el puntaje más bajo y esto pudo darse por varias causas. Se considera que la causa principal fue que la mayoría de las personas que realizaron las pruebas solicitaron una explicación antes de realizar los casos de prueba lo cual consideraron como apoyo técnico.

Otro de las preguntas que obtuvo un puntaje bajo fue la pregunta 1 (¿Cree que usaría esta aplicación frecuentemente?), esto debido a que la encuesta se la realizó a diferentes personas que no estaban relacionadas al campo de la agricultura por lo que su respuesta fue que no utilizarían la aplicación.

La pregunta 10 (¿Necesitó aprender varias cosas antes de poder utilizar la aplicación?) también obtuvo un puntaje bajo. Una de las causas es por la falta de comprensión con respecto a la aplicación y al flujo de esta. Debido a que la encuesta se realizó a personas

externas, se tuvo que dar una explicación antes de realizar los casos de prueba, por lo que se puede justificar los resultados obtenidos.

Una de las preguntas con el puntaje más alto fue la número 2 (¿Encuentra esta aplicación innecesariamente compleja?). Esto se debe a que la aplicación tiene como objetivo principal, mostrar los datos obtenidos de los sensores a los usuarios para un monitoreo de la humedad del suelo por lo que resulta ser una aplicación sencilla a simple vista.

En la Tabla 54 se muestra el puntaje obtenido por cada usuario según la escala SUS.

Usuarios Encuestados	Puntaje SUS
1	82,5
2	100
3	87,5
4	75
5	90
6	92,5
7	87,5
8	80
PROMEDIO	86,87

Tabla 54. Puntajes obtenidos según la escala SUS por cada usuario

Como se puede apreciar en la Tabla 54 el puntaje promedio obtenido es de 86.87. Según se menciona en [52], cuando se obtiene un puntaje por encima de 68, el sistema se considera altamente usable.

En el Anexo 6: Resultados de las Encuestas de Pruebas de Usabilidad se puede observar a mayor detalle los resultados de la Encuesta de Usabilidad.

5.3 Prueba de Cobertura

Para probar la red LoRaWAN diseñada en el presente proyecto, se realizó una prueba de cobertura colocando los sensores en el cultivo de fresas que se encuentra ubicado a una distancia de 750 metros del gateway aproximadamente (Figura 111).



Figura 111. Distancia medida entre el gateway y el cultivo de fresas. Tomada desde Google Maps.

El gateway fue colocado en el interior de una casa para ser conectado al Internet mediante un cable Ethernet. En la Tabla 55 se muestran los puntos de ubicación por GPS de los dispositivos de la red LoRaWAN.

Dispositivo	Punto de ubicación (GPS)
Gateway	-0.172729, -78.337432
Sensor LSE01-01	-0.178292, -78.337621
Sensor LSE01-02	-0.178076, -78.337958

Tabla 55. Puntos de ubicación de los dispositivos de la red LoRaWAN.

Los sensores fueron instalados de manera horizontal a la altura de las raíces de las plantas de fresas aproximadamente a 16 cm de profundidad, como se muestra en la Figura 112.



Figura 112. Sensor instalado en el suelo de manera horizontal.

En la Figura 113 y Figura 114 se muestran los sensores LSE01-01 y LSE01-02 instalados en el cultivo de fresas.



Figura 113. Sensor LSE01-01 instalado en el cultivo de fresas.



Figura 114. Sensor LSE01-02 instalado en el cultivo de fresas.

Después de haber realizado la instalación de los sensores, se verificó la obtención de datos en la aplicación. En la Figura 115 se puede verificar los valores reales obtenidos desde los sensores ya instalados en el cultivo de fresas.



Figura 115. Recepción de datos de los sensores instalados.

Según las pruebas de cobertura realizadas se pudo comprobar una de las características del protocolo LoRaWAN que es lograr comunicación a grandes distancias.

CAPÍTULO 6. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- El conocimiento a profundidad de todo el proceso de cultivo de fresas desde la siembra hasta la cosecha de los frutos facilitó la obtención de requerimientos, así como también se logró comprender de mejor manera las necesidades de los agricultores. Esto facilitó la implementación del proyecto.
- Se obtuvo valores reales de la humedad del suelo en el cultivo de fresas gracias a los sensores instalados. Esto permitió mejorar el proceso de cultivo optimizando recursos de tiempo y agua. También se pudo evitar algunas enfermedades que eran ocasionadas por el exceso o escaso riego en el suelo. Uno de los beneficios conseguidos fue evitar frutos pequeños por falta de agua en el cultivo.
- La tecnología LoRaWAN es muy flexible para la solución de problemas de IoT, aún más en el campo de la agricultura, gracias a sus diversas ventajas como su largo alcance, bajo consumo de energía y larga duración. Siendo así una tecnología económicamente rentable y fácil de implementar.
- El diseño e implementación de la red LoRaWAN en el cultivo de fresas supone un aporte al desarrollo del sector agrícola. Gracias a la implementación de nuevas tecnologías en la agricultura, se puede obtener procesos de cultivo mejorados y optimizados.
- La metodología SCRUM permitió el cumplimiento de los objetivos planteados al inicio. Ayudó a mantener siempre la participación de los interesados logrando así el desarrollo del sistema. En cada etapa del proyecto se obtuvo una retroalimentación de los interesados adquiriendo una gestión de cambios eficiente.
- El uso de la plataforma Firebase en el proceso de autenticación y base de datos agilitó el desarrollo del sistema proporcionando la infraestructura necesaria para la creación de aplicaciones sin necesidad de tener un servidor específico para el proyecto. Gracias a su amplia documentación se logró resolver cualquier inquietud acerca de la plataforma.

- Después de haber implementado el sistema, las pruebas de funcionalidad, usabilidad y cobertura permitieron conocer el cumplimiento de los objetivos planteados al inicio del proyecto. En las pruebas realizadas se obtuvo un resultado satisfactorio en cuanto a la aceptación del sistema.
- En las pruebas de funcionalidad se obtuvo un resultado del 94%, lo que significa que fue posible realizar la mayoría de las funciones de la aplicación sin inconvenientes. Además, se puede mencionar que las personas que probaron la aplicación no necesitaron conocimientos técnicos para lograr los resultados esperados gracias a las interfaces de usuario amigables y el flujo sencillo de la aplicación.
- Las pruebas de usabilidad dieron como resultado un puntaje de 86.87 en la escala de SUS. Esto indica que la aplicación es altamente usable y clara basada en la experiencia del usuario. Al realizar estas pruebas se puede evitar errores antes de que la aplicación web sea puesta en producción.
- Las pruebas de cobertura permitieron conocer el alcance de la tecnología LoRaWAN. Cuando los sensores fueron ubicados en el cultivo a una distancia lejana al gateway se pudo comprobar la correcta conexión entre los equipos. Esto representa un beneficio considerable pues los sensores pueden ser ubicados en cualquier parte del terreno inclusive a kilómetros de distancia del gateway sin perder la conexión.

6.2 Recomendaciones

- Se recomienda analizar las limitaciones de las herramientas utilizadas. En el caso particular de Firebase, se debe tomar en cuenta los costos de los planes de uso disponibles según el análisis de uso de la aplicación.
- Una vez que la aplicación sea puesta en producción, se recomienda analizar las métricas de esta como el uso de memoria, el rendimiento, el tráfico de la red, entre otras. De esta manera se podrá elegir el mejor plan de despliegue en la plataforma Heroku o en cualquier otra herramienta similar a esta.

- Se recomienda considerar una futura implementación haciendo uso de los datos obtenidos de los sensores para lograr un proceso de control automático de riego. Mediante la detección de niveles de humedad se puede generar una respuesta hacia el gateway activando o desactivando conjuntamente la bomba y las válvulas de riego. Esto incrementaría los beneficios hacia el agricultor tales como la reducción de mano de obra, optimización de tiempo y recursos hídricos.
- Se recomienda analizar la implementación de una aplicación móvil que sea más práctica para el uso del agricultor. Esta aplicación puede ofrecer las notificaciones en el celular, en tiempo real sin necesidad de abrir la aplicación.
- Analizar el manejo de los datos en la aplicación, puesto que se puede tener datos innecesarios en la base de datos. Se puede crear una política de gestión de datos que permita eliminar datos antiguos que ya no sean relevantes para el agricultor.

BIBLIOGRAFÍA

- [1] Redacción Tierra y Mar, "El 8% del PIB lo genera el sector agropecuario," *eltelegrafo*, 2018. <https://www.eltelegrafo.com.ec/noticias/economia/4/pib-sector-agropecuario-ecuador> (accessed Nov. 04, 2019).
- [2] T. Useche, C. Jordan, L. Rodriguez, and S. Ronald, "PROTOTIPO DE SOLUCIÓN IoT CON TECNOLOGÍA 'LoRa' EN MONITOREO DE CULTIVOS AGRÍCOLAS," Universidad Distrital Francisco José de Caldas, 2018.
- [3] L. Torres, "Entienda la importancia del protocolo LoRa™ en la Agricultura 4.0," *Blog de Khomp*, 2018. <https://www.khomp.com/es/protocolo-LoRa-na-agricultura-4-0/> (accessed Nov. 04, 2019).
- [4] J. D. Borrero, *Aplicación de la tecnología LoRaWAN en la agricultura*. 2018.
- [5] Universidad de Huelva, "Cómo afectan los factores climatológicos a la producción de la fresa," *DESQBRE Fundación*, 2013. <https://fundaciondescubre.es/noticias/como-afectan-los-factores-climatologicos-a-la-produccion-de-la-fresa/> (accessed Mar. 28, 2020).
- [6] Cámara de Agricultura, "El Cultivo De La Frutilla," *Cámara de Agricultura*, 2016. <http://agroecuador.org/index.php/blog-noticias/item/93-el-cultivo-de-la-frutilla> (accessed Mar. 26, 2020).
- [7] CatSensors, "Soluciones LoRa para agricultura inteligente," *iAgua*, 2019. <https://www.iagua.es/noticias/catsensors/soluciones-LoRa-agricultura-inteligente> (accessed Oct. 21, 2019).
- [8] INEC, "ESTRUCTURA DEL SECTOR AGROPECUARIO, SEGÚN EL ENFOQUE DE LAS CARACTERÍSTICAS DEL PRODUCTOR AGROPECUARIO Y DE LAS UNIDADES DE PRODUCCIÓN AGROPECUARIA," Quito, 2008.
- [9] J. Salazar and S. Silvestre, "Internet de las cosas," *Univ. Católica*, pp. 1–27, 2014.
- [10] V. Andrés, "Study of Precision Agriculture with a Focus on Implementing a Wireless Sensor Network (WSN) for Humidity and Temperature Monitoring in Crops-Case Study Cabalinus Farm Located in Los Rios Province," *Rev. Politécnica-Abril*, vol. 38, no. 1, 2016, [Online]. Available: http://repositorio.puce.edu.ec/bitstream/handle/22000/11112/Articulo_cientifico_Andrés_Vela-Revista_EPN.PDF?sequence=1&isAllowed=y.
- [11] R. Ortega B. and L. Flores M., "Agricultura de precisión: Introducción al manejo sitio-específico," *Inf. Agronómicas del Cono Sur*, vol. 7, pp. 1–5, 2000.
- [12] CatSensors, "Tecnología LoRa y LoRaWAN," *CatSensors*, 2020.

- catsensors.com/es/LoRawan/tecnologia-LoRa-y-LoRawan (accessed Sep. 12, 2020).
- [13] Semtech, "What is LoRa?," *Semtech*, 2020. <https://www.semtech.com/LoRa/what-is-LoRa> (accessed Sep. 18, 2020).
- [14] LoRa Alliance, "What is the LoRaWAN Specification?," *LoRa Alliance*, 2015. <https://LoRa-alliance.org/about-LoRawan> (accessed May 16, 2020).
- [15] LoRa Alliance, "LoRaWAN 1.0.3 specification," *LoRa-Alliance.Org*, no. 1 [Online], Accessible: <https://LoRa-alliance.org/sites/default/files/2018-07/LoRawan1.0.3.pdf>, pp. 1–72, 2018, [Online]. Available: <https://LoRa-alliance.org/sites/default/files/2018-07/LoRawan1.0.3.pdf>.
- [16] The Things Network, "LoRaWAN," *The Things Network, TTN*, 2020. <https://www.thethingsnetwork.org/docs/LoRawan/> (accessed Sep. 12, 2020).
- [17] P. Grossetete, "Cisco @ LoRaWAN™ Technical Overview," *Cisco live*, p. 66, 2018.
- [18] E. Contreras, "Haciendo IoT con LoRa: Capitulo 2.- Tipos y Clases de Nodos," *Medium*, 2017. <https://medium.com/beelan/haciendo-iot-con-LoRa-capitulo-2-tipos-y-clases-de-nodos-3856aba0e5be> (accessed Sep. 12, 2020).
- [19] B. Ray, "NB-IoT frente a LoRa frente a Sigfox," 2018. <https://www.link-labs.com/blog/nb-iot-vs-LoRa-vs-sigfox> (accessed Jan. 27, 2021).
- [20] Adobe Inc, "Adobe XD," *Adobe Inc*, 2020. <https://www.adobe.com/la/products/xd/details.html> (accessed Jul. 20, 2020).
- [21] Lucidchart, "Lucidchart," 2021. <https://www.lucidchart.com/pages/es> (accessed Jan. 24, 2021).
- [22] Diagramas.net, "Draw.io," *diagramas.net*, 2020. <https://app.diagrams.net/> (accessed Jul. 20, 2020).
- [23] MDN web docs, "HTML: Hypertext Markup Language," *MDN web docs*, 2020. <https://developer.mozilla.org/en-US/docs/Web/HTML> (accessed Jun. 16, 2020).
- [24] JavaScript.Info, "An Introduction to JavaScript," *JAVASCRIPT.INFO*, 2020. <https://JavaScript.info/intro> (accessed Jun. 16, 2020).
- [25] MDN web docs, "CSS: Cascading Style Sheets," *MDN web docs*, 2020. <https://developer.mozilla.org/en-US/docs/Web/CSS> (accessed Jun. 17, 2020).
- [26] World Wide Web Consortium (W3C), "What is CSS?," *w3.org*, 2020. <https://www.w3.org/Style/CSS/Overview.en.html> (accessed Jun. 16, 2020).
- [27] Google, "Firebase," *Google*, 2020. <https://firebase.google.com/?hl=es> (accessed Aug. 01, 2020).
- [28] Firebase, "Cloud Firestore," *Firebase Google*, 2020. <https://firebase.google.com/products/firestore> (accessed Jul. 20, 2020).

- [29] Firebase, "Firebase Authentication," *Firebase Google*, 2020. <https://firebase.google.com/docs/auth?hl=es-419> (accessed Jul. 20, 2020).
- [30] Salesforce.com, "What is Heroku?," 2021. <https://www.heroku.com/what> (accessed Jan. 25, 2021).
- [31] Salesforce.com, "Dynos: the heart of the Heroku platform," 2021. <https://www.heroku.com/dynos> (accessed Jan. 25, 2021).
- [32] Semantic UI, "Semantic UI," *Semantic UI*, 2020. <https://semantic-ui.com/> (accessed Jul. 20, 2020).
- [33] Nodejs, "Node JS," *Nodejs*, 2020. <https://nodejs.org/es> (accessed Jul. 20, 2020).
- [34] Expressjs, "Express JS," *Expressjs*, 2017. <https://expressjs.com/> (accessed Jul. 20, 2020).
- [35] socket.io, "Socket.io," 2021. <https://socket.io/docs/v3> (accessed Jan. 21, 2021).
- [36] Atlassian Bitbucket, "Atlassian Bitbucket," *Atlassian Bitbucket*, 2020. <https://bitbucket.org/> (accessed Jun. 18, 2020).
- [37] Microsoft, "Visual Studio Code," *Microsoft*, 2020. <https://code.visualstudio.com/>.
- [38] Google, "Google Earth," *Google*, 2020. <https://earth.google.com/> (accessed Aug. 27, 2019).
- [39] InfoAgro, "El cultivo de la Fresa." https://www.infoagro.com/documentos/el_cultivo_fresa.asp (accessed Apr. 02, 2020).
- [40] P. Undurraga and S. Vargas, "Manual de Frutilla," Chillán, 2013.
- [41] D. Bankov, E. Khorov, and A. Lyakhov, "On the limits of LoRaWAN channel access," *Proc. - 2016 Int. Conf. Eng. Telecommun. EnT 2016*, pp. 10–14, 2017, doi: 10.1109/EnT.2016.9.
- [42] Digebis, "Cómo plantar y cuidar fresas," 2015. <http://www.digebis.com/blog/como-plantar-y-cuidar-fresas/> (accessed Mar. 25, 2020).
- [43] T. Jardón, "Plagas y Enfermedades De La Fresa o Frutilla," 2017. <https://www.lahuertinadetoni.es/plagas-y-enfermedades-de-la-fresa-o-frutilla/#:~:text=Mancha Purpura o Viruela del,manchas circulares de color rojo.>
- [44] MeterGroup, "The researcher's complete guide to soil moisture," *MeterEnvironmental*, 2020. <https://www.metergroup.com/environment/articles/the-researchers-complete-guide-to-soil-moisture/> (accessed Jul. 24, 2020).
- [45] L. Zotarelli, M. Dukes, and M. Paranhos, "Minimum Number of Soil Moisture Sensors for Monitoring and Irrigation Purposes," *HS1222, Hortic. Sciecne Dep. UF/IFAS Ext.*, pp. 2–5, 2013.
- [46] J. Enciso, D. Porter, and X. Péries, "Uso de sensores de humedad en suelo para

mejorar el manejo del riego en cultivos de invernadero," *Mejor. en la Efic. del uso agua y Fertil. en Agric.*, vol. 39, no. 5, pp. 107–132, 2018, [Online]. Available: <https://aglifesciences.tamu.edu/baen/wp-content/uploads/sites/24/2017/01/E-618S-Irrigation-Monitoring-with-Soil-Water-Sensors.-Spanish-Version.pdf>.

- [47] Dragino, "Dragino," *Dragino*, 2020. <https://www.dragino.com/index.php/about/concept> (accessed Jul. 27, 2020).
- [48] Dragino, "LPS8 LoRaWAN Gateway User Manual," pp. 1–32, 2019.
- [49] Dragino, "LoRaWAN Soil Moisture & EC Sensor User Manual," pp. 1–30, 2020.
- [50] J. R. Urban, "SOILS: THE MEASURE OF MOISTURE," *Landscape Architecture Magazine*, Jan. 2014.
- [51] L. Pitts, "Monitoring Soil Moisture for Optimal Crop Growth Larry Pitts," 2016. https://observant.zendesk.com/hc/en-us/articles/208067926-Monitoring-Soil-Moisture-for-Optimal-Crop-Growth?mobile_site=true (accessed Feb. 04, 2021).
- [52] usability.gov, "System Usability Scale (SUS)," 2021. <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html> (accessed Jan. 29, 2021).

ANEXOS

Anexo 1: Mockups de la Aplicación Web

Se adjunta el enlace para poder visualizar los mockups de la aplicación web realizados en Adobe XD:

<https://xd.adobe.com/view/bb007a62-447e-4f67-8a5a-1991ce0065e8-ee53/?fullscreen>

Enlace de descarga de los mockups de la aplicación web:

<https://1drv.ms/u/s!Akshnm553GEhZwhOPLzOk1RwZStrg?e=CTGteQ>

Anexo 2: Resultados de los Casos de Prueba

Enlace de descarga de los resultados de los casos de prueba de las pruebas de funcionalidad:

<https://1drv.ms/x/s!Akshnrn553GEhZwUxSxbxwWV-K6V4g?e=mu0uw5>

Anexo 3: Encuesta de Pruebas de Funcionalidad

Enlace de descarga de la encuesta de las pruebas de funcionalidad:

<https://1drv.ms/b/s!Akshnm553GEhZwfg4yUF3uKgmNriw?e=LjnKde>

Anexo 4: Encuesta de Pruebas de Usabilidad

Enlace de descarga de la encuesta de las pruebas de usabilidad:

<https://1drv.ms/b/s!Akshnm553GEhZwjX1IWKY6bzXHFkw?e=kO4jrj>

Anexo 5: Resultados de las Encuestas de Pruebas de Funcionalidad

Enlace de descarga de la encuesta de las pruebas de funcionalidad:

<https://1drv.ms/b/s!Akshnm553GEhZw11tQRvbOpw9w1FQ?e=xxZHfi>

Anexo 6: Resultados de las Encuestas de Pruebas de Usabilidad

Enlace de descarga de la encuesta de las pruebas de usabilidad:

<https://1drv.ms/b/s!Akshnm553GEhZwwXI4oSSQFXpE5Hg?e=ZQ1LQf>

Anexo 7: Video demostrativo de la funcionalidad de la Aplicación Web

Enlace de descarga del video:

https://1drv.ms/v/s!Akshnm553GEhZxSR_E0nhF2hnnx7g?e=EQ8orm