

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**

Casco Inteligente para la Prevención de Golpes de Calor

Corporación Universitaria Remington.  
Facultad Ingeniería de sistemas.  
Tecnología en Desarrollo de software e Ingeniería de sistemas

Luz Elena Romero Acosta  
Andrés Quiroga Cortés  
Jailer Cuadrado Vanegas

Nombre del Tutor del trabajo de grado.  
Gloria Amparo Lora

Proyecto de grado  
2025

### Dedicatoria

El presente trabajo se lo dedicamos primeramente a Dios que nos ha permitido llegar hasta este momento de nuestro camino en la educación, abriendo nuestras mentes y brindándonos la sabiduría para seguir creciendo cada día como personas y profesionales. También agradecemos a nuestras familias, quienes nos apoyaron en cada una de las fases del proyecto con su compañía, paciencia y comprensión.

### Agradecimientos

Queremos expresar nuestro más profundo agradecimiento a Gloria Amparo Lora y Piedad Metaute quienes fueron nuestras tutoras en el trabajo de grado porque nos aportan su experiencia, exigencia, comprensión y paciencia, factores indispensables para contribuir en el complejo y gratificante camino de esta investigación. Su guía constante y su refuerzo en fortalecer nuestras habilidades nos han motivado a alcanzar metas que no imaginábamos al inicio del camino.

Un sincero agradecimiento mutuo también a mis amigos y compañeros que estuvimos en este trabajo, por todos los momentos que compartimos de estrés y alegrías durante este largo y retador camino. El apoyo, confianza, respeto a la opinión, compromiso y cumplimiento con cada una de las tareas propuestas fue el aporte más importante que cada uno añadió para ser hoy de esta meta una realidad.

## Tabla de contenido

1. Resumen.....	10
2. Palabras clave.....	11
INTRODUCCIÓN .....	11
3. Marco teórico .....	12
3.1 Seguridad y salud en el trabajo .....	12
3.2 Oleadas de Calor .....	12
3.3 Enfermedades por Golpes de calor .....	13
3.4 Tipos de Empleados .....	15
3.5 Regiones de Colombia Afectadas por el calor .....	16
3.6 Sistemas de Monitoreo.....	17
3.7 Microcontrolador .....	17
3.8 Sensores .....	17
3.9 Sistemas de geolocalización .....	17
3.10 Aplicaciones móviles .....	18
3.11 Metodología de desarrollo de Software .....	18
3.12 Metodología ágil Scrum.....	18
4. Problema .....	19
4.1 Planteamiento del problema.....	19
4.2 Pregunta de Investigación .....	20
5. Objetivos .....	20
5.1 Objetivo General .....	20
5.2 Objetivos específicos .....	20
6. Metodología .....	21
6.1. Fases análisis.....	21
6.2 Formato de la encuesta: .....	22
6.3 Análisis de la encuesta:.....	23
7. Conclusión de la encuesta.....	27
8. Historias de usuarios .....	27
8.1 Historias de Usuario Prioritarias (MVP - Versión 1.0).....	28
8.2 Historias de usuarios ajustadas .....	29
9. Alcance del sistema propuesto en términos de (entradas, procesos y salidas) .....	30
10. Alcance del Sistema.....	31
11. Nombre de la aplicación .....	31
12. Cronograma de Actividades.....	32
13. Análisis de riesgos .....	33
14. Análisis de Requisitos.....	36
14.1 Actores del sistema .....	36
14.2 Tabla general para casos de uso.....	37
14.3 Requisitos de usuario (RU).....	42
14.4 Requisitos funcionales (RF).....	44
14.5 Requisitos no funcionales .....	55

15. Escenarios de los Casos de Uso .....	60
16. Especificación del Requisito .....	95
16.1 Diagrama de caso de uso principal .....	96
16.2 Diagramas casos de uso extendidos .....	97
17. Mapa de Navegación.....	108
17.1 Tipos de mapas de navegación web .....	108
17.2 Estructura jerárquica .....	108
17.3 Estructura web .....	108
17.4 Estructura compuesta .....	109
17.5. Mapa de Navegación.....	110
19. Recursos (Hardware, Software, Talento Humano).....	111
19.1 Recurso Hardware.....	111
19.2 Recurso software.....	112
19.3 Recurso talento Humano.....	113
20. Modelo Entidad Relación .....	114
21. Diagrama de clases .....	115
22. Diagrama de Arquitectura.....	117
23 Diagrama de Secuencia.....	118
24. Justificación de Hardware y Software .....	119
25. Diseño .....	119
25.1 Mockups.....	120
26. Fase de Desarrollo.....	139
26.1 Estructura del Proyecto .....	139
26.2 Configuración de Entornos .....	140
26.3 Código Fuente Relevante .....	146
26.4 Componentes.....	157
26.5 Integración de Componentes.....	163
26.6 Evidencia de Procesos Principales .....	167
Proceso Registro de Usuario .....	167
Proceso inicio de sesión de Usuario.....	171
Proceso perfil SST .....	173
Proceso perfil administrador .....	175
26.7 Documentación Técnica.....	179
27. Manual de Usuario .....	179
28. Manual del Sistema.....	206
29. Fase de Pruebas.....	225
30. Resultados y Discusión .....	229
Conclusiones .....	230
Referencias.....	232

## Lista de tablas

(Incluya esta sección sólo si aplica para su trabajo de grado)

?	TABLA 1 COMPARATIVA DE LOS RAYOS UVA Y UVB.....	15
?	TABLA 2 ALCANCE DEL SISTEMA.....	30
?	TABLA 3 ANÁLISIS DE RIESGOS .....	35
?	TABLA 4 ACTORES DEL SISTEMA .....	36
?	TABLA 5 GENERAL PARA CASOS DE USO .....	37
?	TABLA 6 TEMPERATURA AMBIENTE Y HUMEDAD.....	38
?	TABLA 7 TEMPERATURA CORPORAL Y REGISTRO DE TIPS.....	39
?	TABLA 8 UBICACIÓN DE OBRA Y ASIGNAR CASCO.....	40
?	TABLA 9 DEFINIR RANGOS DE TEMPERATURA .....	40
?	TABLA 10 PERFIL DE USUARIO .....	41
?	TABLA 11 USUARIO .....	41
?	TABLA 12 GESTIÓN CONSULTA .....	42
?	TABLA 13 GESTIÓN INFORMES .....	42
?	TABLA 14 REQUISITOS DE USUARIOS .....	44
?	TABLA 15 REQUISITOS FUNCIONALES.....	54
?	TABLA 16 REQUISITOS NO FUNCIONALES .....	55
?	TABLA 17 CONFIABILIDAD.....	56
?	TABLA 18 PERFORMANCE .....	56
?	TABLA 19 RESTRICCIONES DE DISEÑO .....	57
?	TABLA 20 SEGURIDAD.....	57
?	TABLA 21 DOCUMENTACIÓN DE USUARIO Y SISTEMAS DE AYUDA.....	58
?	TABLA 22 INTERFAZ DE USUARIO .....	58
?	TABLA 23 INTERFACES DE COMUNICACIÓN .....	59
?	TABLA 24 ESCENARIOS DE LOS CASOS DE USO .....	63
?	TABLA 25 CONSULTAR TRABAJADOR.....	65
?	TABLA 26 MODIFICAR TRABAJADOR.....	67
?	TABLA 27 GUARDAR TRABAJADOR.....	68
?	TABLA 28 INHABILITAR TRABAJADOR.....	70
?	TABLA 29 CREAR REGISTRO DE TIPS .....	71
?	TABLA 30 CONSULTAR REGISTRO DE TIPS .....	73
?	TABLA 31 MODIFICAR REGISTRO DE TIPS .....	74
?	TABLA 32 GUARDAR REGISTRO DE TIPS .....	76
?	TABLA 33 INHABILITAR REGISTRO DE TIPS .....	77
?	TABLA 34 CREAR TEMPERATURA CORPORAL.....	78
?	TABLA 35 CONSULTAR TEMPERATURA CORPORAL.....	80
?	TABLA 36 MODIFICAR TEMPERATURA CORPORAL.....	82
?	TABLA 37 GUARDAR TEMPERATURA CORPORAL .....	83
?	TABLA 38 INHABILITAR TEMPERATURA CORPORAL.....	85

?	TABLA 39 INFORME DE USUARIOS REGISTRADOS.....	86
?	TABLA 40 INFORME DE ACTIVACIÓN DE ALERTAS CRÍTICAS .....	88
?	TABLA 41 INFORME ESTADO DE LA TEMPERATURA DE LOS USUARIOS .....	89
?	TABLA 42 CONSULTAR UBICACIÓN EN PANEL DE MONITOREO .....	91
?	TABLA 43 CONSULTAR TIPS EN EL BANNER DE LA APP .....	93
?	TABLA 44 CONSULTAR HISTORIAL DE ALERTAS .....	95

### Lista de figuras

(Incluya esta sección sólo si aplica para su trabajo de grado)

?	FIGURA 1 RESULTADOS DE LA RESPUESTA _____	23
?	FIGURA 2 RESULTADOS DE LA RESPUESTA _____	24
?	FIGURA 3 RESULTADOS DE LA RESPUESTA _____	25
?	FIGURA 4 RESULTADOS DE LA RESPUESTA _____	26
?	FIGURA 5 DIAGRAMA PRINCIPAL CASOS DE USO _____	96
?	FIGURA 6 CASOS DE USOS EXTENDIDOS (TRABAJADOR) _____	97
?	FIGURA 7 CASOS DE USO EXTENDIDO (TEMPERATURA AMBIENTE Y HUMEDAD)	98
?	FIGURA 8 CASOS DE USO EXTENDIDO (TEMPERATURA CORPORAL) _____	99
?	FIGURA 9 CASOS DE USO EXTENDIDO (REGISTRO DE TIPS) _____	100
?	FIGURA 10 CASOS DE USO EXTENDIDO (UBICACIÓN DE OBRA) _____	101
?	FIGURA 11 CASOS DE USO EXTENDIDO (ASIGNAR CA) _____	102
?	FIGURA 12 CASOS DE USO EXTENDIDO (DEFINIR RANGOS DE TEMPERATURA) _	103
?	FIGURA 13 CASOS DE USO EXTENDIDO (PERFIL DE USUARIO) _____	104
?	FIGURA 14 CASOS DE USO EXTENDIDO (USUARIO) _____	105
?	FIGURA 15 CASOS DE USO EXTENDIDO (GESTIÓN CONSULTA) _____	106
?	FIGURA 16 CASOS DE USO EXTENDIDO (GESTIÓN INFORME) _____	107
?	FIGURA 17 MAPA NAVEGACIÓN _____	107
?	FIGURA 18 MODELO ENTIDAD RELACIÓN _____	114
?	FIGURA 19 DIAGRAMA DE CLASES _____	116
?	FIGURA 20 DIAGRAMA DE ARQUITECTURA _____	117
?	FIGURA 21 DIAGRAMA DE SECUENCIA _____	118
?	FIGURA 22 PANTALLA DE INICIO (USUARIO) _____	120
?	FIGURA 23 PANTALLA DE INICIO (USUARIO) _____	120
?	FIGURA 24 LOGIN (USUARIO) _____	121
?	FIGURA 25 LOGIN (USUARIO) _____	121
?	FIGURA 26 PANTALLA REGISTRO NUEVO USUARIO _____	122
?	FIGURA 27 PANTALLA REGISTRO NUEVO USUARIO _____	122
?	FIGURA 28 PANTALLA DE RECUPERACIÓN CONTRASEÑA (USUARIO) _____	123
?	FIGURA 29 PANTALLA DE RECUPERACIÓN CONTRASEÑA (USUARIO) _____	123
?	FIGURA 30 PANTALLA CONFIRMACIÓN ENVIO DE CORREO (USUARIO) _____	124

?	FIGURA 31 PANTALLA CONFIRMACIÓN ENVIO DE CORREO (USUARIO)	124
?	FIGURA 32 PANTALLA INFORMACIÓN USUARIO	125
?	FIGURA 33 PANTALLA INFORMACIÓN USUARIO	125
?	FIGURA 34 PANTALLA CERRAR SESIÓN (USUARIO)	126
?	FIGURA 35 PANTALLA CERRAR SESIÓN (USUARIO)	126
?	FIGURA 36 PANTALLA DE INICIO (SST)	127
?	FIGURA 37 PANTALLA DE INICIO (SST)	127
?	FIGURA 38 PANTALLA VISTA MONITOREO EMPLEADOS (SST)	128
?	FIGURA 39 PANTALLA VISTA MONITOREO EMPLEADOS (SST)	128
?	FIGURA 40 PANTALLA RANGOS DE TEMPERATURA (SST)	129
?	FIGURA 41 PANTALLA RANGOS DE TEMPERATURA (SST)	129
?	FIGURA 42 PANTALLA INFORMES Y CONSULTAS (SST)	130
?	FIGURA 43 PANTALLA INFORMES Y CONSULTAS (SST)	130
?	FIGURA 44 PANTALLA DE INICIO (ADMINISTRADOR)	131
?	FIGURA 45 PANTALLA DE INICIO (ADMINISTRADOR)	131
?	FIGURA 46 PANTALLA GESTIÓN DE ROLES (ADMIN)	132
?	FIGURA 47 PANTALLA GESTIÓN DE ROLES (ADMIN)	132
?	FIGURA 48 PANTALLA ASIGNACIÓN DISPOSITIVO (ADMIN)	133
?	FIGURA 49 PANTALLA ASIGNACIÓN DISPOSITIVO (ADMIN)	133
?	FIGURA 50 PANTALLA GESTIÓN DE TIPS (ADMIN)	134
?	FIGURA 51 PANTALLA GESTIÓN DE TIPS (ADMIN)	134
?	FIGURA 52 PANTALLA RANGOS DE TEMPERATURA (ADMIN)	135
?	FIGURA 53 PANTALLA RANGOS DE TEMPERATURA (ADMIN)	135
?	FIGURA 54 PANTALLA HISTORIAL GOLPES DE CALOR (ADMIN)	136
?	FIGURA 55 PANTALLA HISTORIAL GOLPES DE CALOR (ADMIN)	136
?	FIGURA 56 PANTALLA DE INFORMES Y CONSULTAS (ADMIN)	137
?	FIGURA 57 PANTALLA DE INFORMES Y CONSULTAS (ADMIN)	137
?	FIGURA 58 PANTALLA EMPLEADOS REGISTRADOS	138
?	FIGURA 59 PANTALLA EMPLEADOS REGISTRADOS	138
?	FIGURA 60 ESTRUCTURA DEL PROYECTO	140
?	FIGURA 61 IDE USADO	140
?	FIGURA 62 DEPENDENCIAS INSTALADAS	141
?	FIGURA 63 CONFIGURACIÓN DE FIREBASE	142
?	FIGURA 64 FAIREBASE JSON	143
?	FIGURA 65 GOOGLE -SERVISE JISON	144
?	FIGURA 66 CONFIGURACIÓN DE ARDUINO	145
?	FIGURA 67 MAIN	146
?	FIGURA 68 MODELS	147
?	FIGURA 69 CONTROLLER	148
?	FIGURA 70 MODEL USUARIOREPOSITORY	149

?	FIGURA 71 AUTENTICACION _____	150
?	FIGURA 72 VISTA PRINCIPAL _____	151
?	FIGURA 73 PANTALLA SST _____	152
?	FIGURA 74 VISTA ADMIN _____	153
?	FIGURA 75 INICIO DE SESIÓN _____	154
?	FIGURA 76 REGISTRO _____	155
?	FIGURA 77 RUTAS _____	156
?	FIGURA 78 INTEGRACIÓN HARDWARE _____	159
?	FIGURA 79 MEDICIÓN DE TEMPERATURA _____	163
?	FIGURA 80 ALERTA NARANJA _____	164
?	FIGURA 81 ALERTA ROJA _____	165
?	FIGURA 82 NOTIFICACIÓN A SST _____	166
?	FIGURA 83 REGISTRO USUARIO _____	164
?	FIGURA 84 INICIO DE SESIÓN _____	167
?	FIGURA 85 BASE DE DATOS FIREBASE _____	168
?	FIGURA 86 BASE DE DATOS FIREBASE _____	168
?	FIGURA 87 RECUPERAR CONTRASEÑA _____	169
?	FIGURA 88 LINK RECUPERAR _____	169
?	FIGURA 89 CAMBIO DE CONTRASEÑA _____	170
?	FIGURA 90 INGRESO _____	170
?	FIGURA 91 VISTA PRINCIPAL _____	171
?	FIGURA 92 INICIO DE SESIÓN _____	171
?	FIGURA 94 CIERRE DE SESIÓN _____	172
?	FIGURA 93 PERFIL USUARIO _____	172
?	FIGURA 95 PERFIL SST _____	173
?	FIGURA 96 MONITOREO _____	173
?	FIGURA 97 TEMPERATURA _____	174
?	FIGURA 98 INFORMES Y CONSULTAS _____	174
?	FIGURA 99 PERFIL ADMINISTRADOR _____	175
?	FIGURA 100 ROLES _____	175
?	FIGURA 101 TIPS _____	176
?	FIGURA 102 ASIGNACIÓN DISPOSITIVO _____	176
?	FIGURA 103 RANGOS TEMPERATURA _____	177
?	FIGURA 104 HISTORIALES GOLPES DE CALOR _____	177

## 1. Resumen

En este proyecto damos a conocer un casco inteligente para prevenir riesgos en los trabajadores del gremio de la construcción, que están expuestos a oleadas de calor. El estar expuestos constantemente puede causar problemas de agotamiento, deshidratación, cáncer en la piel y golpes de calor. Debido a esto planteamos un sistema que permite monitorear en tiempo real la temperatura corporal, temperatura ambiente, geolocalización integrados en un casco.

El propósito principal del casco es que pueda generar alertas en tiempo real al personal de SST y trabajadores cuando el calor represente un riesgo para la salud debido al calor extremo. Esto nos permite reaccionar de manera oportuna para evitar accidentes laborales. Es una aplicación sencilla de usar, donde pueden visualizar las alertas relacionadas con el estado de salud en tiempo real de cada trabajador y posteriormente acceder a un historial de alertas recibidas. Esto contribuye a mejorar las condiciones laborales y fomentar una cultura de prevención.

Este estudio parte de entender como las olas de calor afectan a los trabajadores que están expuestos constantemente a ellas, a partir de encuestas realizadas se demostró la necesidad de crear un sistema de monitoreo que ayude a proteger el bienestar de los trabajadores.

Para finalizar, este casco da una solución buena para los trabajadores que están siempre bajo el sol en el trabajo. Su diseño trata no solo de bajar los riesgos del calor sino también ayudar a un ambiente laboral más seguro y que piensa en la salud de los trabajadores.

## 2. Palabras clave

Salud, golpe de calor, sensores, temperatura, software.

## INTRODUCCIÓN

Hemos sido testigos de cómo la radiación solar en los últimos años ha causado grandes efectos sobre el cambio climático, generando grandes problemas en aquellas personas que trabajan diariamente expuestas al sol, uno de los riesgos más notorios ha sido las fuertes oleadas de calor. esto es un problema que día a día es más Fuerte, lo cual ha generado consecuencias a los trabajadores en el área de construcción debido a que la mayor parte de su jornada laboral estos la pasan directamente expuestos a los rayos del sol. Su salud puede verse afectada de diferentes formas, investigando encontramos que diferentes estudios han demostrado que estas condiciones pueden causar desde agotamiento físico, deshidratación, golpes de calor e incluso cáncer de piel. Según la Organización Mundial de la Salud y el Instituto Nacional de Seguridad y Salud en el Trabajo comparten en que este tipo de exposición implican un riesgo laboral, el cual debe ser tratado de manera preventiva. (Organización Mundial de la Salud, 2024).

Debido a estas situaciones, la tecnología se transforma en una herramienta, para disminuir peligros y brindar una mayor protección a la salud de los trabajadores. Una de las opciones es el desarrollo de dispositivos portables que permitan monitorear en tiempo real las condiciones térmicas del ambiente y del propio trabajador. Por lo tanto, en este proyectó se propone el desarrollo de un casco inteligente, el cual puede enviar alertas en tiempo real ya que este está equipado con sensores de temperatura corporal, humedad, ambiental y módulo de geolocalización.

El desarrollo del prototipo contiene un microcontrolador ESP32, un módulo GPS NEO-6M, sensores como el DTH11, el MLX90614, utilizados en entornos de monitoreo ambiental y de salud. Asimismo, basado bajo el lenguaje de programación Dart, Flutter y Firebase lo que permite desarrollar una aplicación móvil multiplataforma que recopile, visualice y almacene los datos en tiempo real, brindando así herramientas tanto al trabajador como al de SST y al administrador de seguridad laboral para actuar de manera oportuna.

### **3. Marco teórico**

La tecnología vestible se ha empezado a usar en varias industrias como forma de vigilar la salud de las personas mientras trabajan. Para el caso de construcción, donde los trabajadores están expuestos a diferentes riesgos ambientales todo el día, esta tecnología sirve para detectar cuando algo anda mal antes de que pase algo serio (García, 2024). Dice que, si se integran sensores a los equipos de protección, se pueden identificar condiciones peligrosas a tiempo y responder más rápido.

El calor extremo es un riesgo grande para quienes trabajan en construcción, sobre todo cuando hay temporadas de mucho sol o altas temperaturas. Pasar horas bajo estas condiciones trae problemas como fatiga, falta de agua en el cuerpo y hasta complicaciones cardíacas. Si hay un sistema que monitoree la temperatura y avise cuando están en niveles críticos, se pueden prevenir estos accidentes actuando antes de que sea tarde.

#### **3.1 Seguridad y salud en el trabajo**

El Sistema de Gestión de Seguridad y Salud en el Trabajo (SG-SST) tiene como objetivo prevenir enfermedades que se pueden causar por las condiciones laborales, se basa en un modelo estructurado el cual consiste en el desarrollo de un proceso progresivo, que promueve la mejora continua, lo cual incluye desde su política y la organización interna, hasta la planificación, la aplicación, la evaluación, la auditoría y las acciones de mejora con el fin de anticiparse y controlar los riesgos que puedan afectar la seguridad y la salud en los espacios laborales. Además, se busca mejorar las condiciones laborales y el ambiente en el trabajo, promoviendo el bienestar físico, mental y social de los empleados (Minambiente, 2022).

#### **3.2 Oleadas de Calor**

La definición más amplia de ola de calor es un periodo de tiempo anómalamente caluroso. Durante varios días seguidos, las temperaturas se mantienen mucho más altas de lo normal para una región determinada. Se trata por tanto de un fenómeno meteorológico extremo. Corresponde a una masa de aire muy cálidas que se instalan sobre una zona

amplia, haciendo que el ambiente se vuelva sofocante, especialmente si hay mucha humedad. (Gobierno de Mexico, 2025).

### 3.3 Enfermedades por Golpes de calor

¿Cómo la radiación solar afecta la salud de los trabajadores?

La exposición a la radiación solar en el entorno laboral es un tema de gran relevancia en materia de seguridad y salud. Cuando trabajamos al aire libre, estamos expuestos a los rayos ultravioleta del sol, los cuales pueden tener efectos perjudiciales en nuestra salud como:

- Quemaduras solares. Una excesiva exposición al sol puede causar quemaduras en la piel, especialmente en áreas expuestas como la cara, los brazos y las piernas. Estas quemaduras pueden variar en gravedad, desde enrojecimiento y sensibilidad hasta ampollas y descamación de la piel.
- Envejecimiento prematuro de la piel. La radiación UV, especialmente los rayos UVA, puede acelerar el envejecimiento de la piel. Esto se manifiesta en forma de arrugas, manchas oscuras y una pérdida de elasticidad de la piel.
- Daño ocular. Una exposición prolongada al sol sin gafas de protección frente los rayos UV puede causar quemaduras en la córnea, inflamación y aumentar el riesgo de cataratas.
- Deshidratación y golpe de calor. Trabajar bajo el sol puede llevar a una mayor pérdida de líquidos, lo que puede derivar en una deshidratación. La excesiva exposición solar y la falta de hidratación aumentan el riesgo de sufrir un golpe de calor, el cual puede poner en riesgo la vida. En 2022, un trabajador de 60 años murió mientras realizaba tareas de limpieza en las calles de Madrid y otro falleció mientras trabajaba en el campo por un golpe de calor.
- Aumento del riesgo de cáncer de piel. La radiación UV es un factor de riesgo para el desarrollo de diferentes tipos de cáncer de piel, como el carcinoma de células escamosas y el melanoma. La Asociación Española contra el Cáncer (AECC) enfatiza la importancia de la protección solar para prevenir el cáncer de piel. Además, recomienda utilizar ropa adecuada que cubra la piel, aplicar protector solar con el factor de protección necesario, utilizar gafas con protección UV y buscar sombra durante las horas de mayor intensidad solar (García, 2024).

- Foto envejecimiento: se refiere al envejecimiento de la piel causado por la exposición a la luz. El prefijo 'foto-significa relativo a la luz', y en el caso del fotoenvejecimiento se refiere específicamente a la luz ultravioleta B (UVB) que emite el sol (Park, 2024)
- Hiperpigmentación: La hiperpigmentación, una afección caracterizada por el oscurecimiento de ciertas áreas de la piel, afecta a personas de todos los tonos y tipos de piel. Si bien no suele ser peligrosa, puede afectar la autoestima, especialmente cuando es visible o afecta áreas extensas del cuerpo. Eritema (enrojecimiento y dolor) o ampollas por exposición prolongada a UVB (Obagi medical, 2024)
- Daño en el ADN de la piel por los rayos UV puede llevar a la formación de células cancerosas con el tiempo, ya que las mutaciones se acumulan y pueden provocar el desarrollo de diversos tipos de cáncer de piel, incluido el melanoma. Este daño se produce porque los rayos UV, que son una forma de radiación carcinógena, dañan directamente el ADN de las células cutáneas, lo que puede manifestarse como envejecimiento prematuro o incluso quemaduras solares, aun cuando no haya síntomas inmediatos. (Worldwide Cancer Research, 2025).
- Fotoqueratitis: se refiere a un daño ocular temporal y doloroso causado por la exposición a la luz ultravioleta (UV), como la del sol y su reflejo, los arcos de soldadura y las luces utilizadas en medicina. Los tratamientos incluyen remedios caseros como compresas frías y analgésicos de venta libre. (Cleveland Clinic, 2025).
- Catarata: por el sol es un término para referirse a la catarata>> causada o acelerada por la exposición prolongada a la radiación ultravioleta (UV) del sol. Esta radiación daña las proteínas del cristalino del ojo>>, volviéndolo opaco y causando visión borrosa. La principal medida preventiva es proteger los ojos con gafas de sol que bloqueen los rayos UV y sombreros, especialmente al pasar tiempo al aire libre. (Clinica Baviera, s.f.)

- El pterigión: (también conocido como carnosidad, pterigium, uña o palmera en el ojo) consiste en un crecimiento anormal de tejido con forma triangular que se extiende desde la conjuntiva (membrana transparente que cubre la esclerótica, la parte blanca del ojo) hasta la córnea (superficie anterior y transparente del ojo) (Clinica Baviera, s.f.)
- Cáncer del ojo es una malignidad que comienza y crece en el ojo. Una malignidad es un grupo de células cancerosas. No son células típicas que se desarrollan rápidamente y en forma descontrolada. Pueden difundirse a otras partes del organismo o pueden invadir o destruir tejidos. (American Academy of Ophthalmology, 2024)

DIFERENCIAS ENTRE UVA Y UVB	
UVA	UVB
Dañan las células internas en la capa superior de la piel, incluida la dermis	Dañan las células en la capa superior de la piel
Tu Piel se broncea de Inmediato	Tu Piel tarda en broncearse
Causan envejecimiento prematuro, arrugas, algunos tipos de Cáncer de piel	Causan cáncer de piel, pueden contribuir al envejecimiento prematuro
Representa aproximadamente el 95% de los rayos solares	Representa aproximadamente el 5% de los rayos solares
Pueden atravesar ventanas y nubes	No atraviesan ventanas y es más probable de las nubes los filtren

- **Tabla 1 comparativa de los rayos UVA y UVB**

### 3.4 Tipos de Empleados

Existen varias ocupaciones que implican una mayor exposición al sol debido a la naturaleza de las tareas realizadas al aire libre. Estos son algunos de los trabajos con mayor riesgo de exposición solar:

- construcción: Estos pasan largas horas expuestos directamente al sol y con la llegada del verano aumentan los riesgos teniendo en cuenta que la mayoría de las obras se realizan en este periodo, el sector de la construcción registra una alta incidencia de enfermedades relacionadas con la exposición solar, como quemaduras, daños en la piel y un aumento del cáncer de piel.
- Pesca: Los trabajadores dedicados a la pesca también son un factor importante a tener en cuenta ante los golpes, estos fuera de encontrarse la mayoría de las ocasiones en mar abierto expuestos directamente a los rayos del sol también se le suma una mayor exposición a los rayos solares debido a la reflexión de estos en la superficie del agua.
- Mantenimiento de espacios verdes y limpieza de calles: A pesar de ser trabajos que no siempre son expuestos directamente al sol se deben tener muy en cuenta, debido a todo el calor corporal que se produce más la sensación térmica del ambiente y la mala hidratación hasta el no uso de la ropa adecuada influye en que se genere un golpe de calor en el mayor de los casos (Garcia, 2024).

### 3.5 Regiones de Colombia Afectadas por el calor

Montería Córdoba, llegó a 37,8 °C

- Cúcuta Norte de Santander, alcanzó los 39,5 °C
- Neiva Huila, con una temperatura de 40,5 °C
- Jerusalén Cundinamarca, con una temperatura de 43,4 °C
- Prado Tolima, que alcanzó los 42 °C
- Guamo Tolima, con 41,1 °C

Estos municipios han registrado temperaturas iguales o superiores a los 38 °C en otros lugares del país. Entre ellos destacan:

- Hatonuevo, La Guajira, con 38,5 °C.
- Villavieja, Huila, con 41 °C.
- Valledupar, Cesar, con 38,8 °C.
- Capitanejo, Santander, con 38,6 °C.
- Armero, Tolima, con 39,6 °C.

- Ataco, Tolima, con 39,8 °C.
  - Natagaima, Tolima, que alcanzó los 43,2 °C.
- (PUBLICACIONES SEMANA S.A., 2024)

### **3.6 Sistemas de Monitoreo**

Un sistema de monitoreo es un conjunto de funcionalidades tanto de software como de hardware que al integrarse recopilan y analizan datos de forma continua lo que permite realizar un seguimiento de procesos en tiempo real, ayudando a identificar problemas de forma temprana y partir de ahí a la toma de una decisión. (Softplan, 2024)

### **3.7 Microcontrolador**

Un microcontrolador es un dispositivo electrónico que puede ser programado para realizar las acciones o instrucciones que nosotros deseemos. Este instrumento funciona de manera similar a como lo haría un ordenador, pero tiene un tamaño menor, un coste más bajo y suele utilizarse para tareas con enfoques más específicos. (Universidad Europea, 2024)

### **3.8 Sensores**

Un sensor es un dispositivo que detecta el cambio en el entorno y responde a alguna salida en el otro sistema. Un sensor convierte un fenómeno físico en un voltaje analógico medible (o, a veces, una señal digital) convertido en una pantalla legible para humanos o transmitida para lectura o procesamiento adicional. (Smith, 2025)

### **3.9 Sistemas de geolocalización**

Un sistema de geolocalización es una solución de la tecnología de la información que determina la ubicación de un objeto en un entorno físico (geo-espacial) o virtual (Internet). A menudo, el objeto es una persona que quiere utilizar un servicio basado en la ubicación, mientras mantiene su privacidad. (Evaluando Software, 2025)

### **3.10 Aplicaciones móviles**

Las aplicaciones móviles o apps son un tipo de software de aplicación, es decir, consisten en programas que se añaden al sistema para poder llevar a cabo diferentes operaciones. Sin embargo, ambos términos no deben confundirse, ya que también constituyen piezas de software de aplicación los programas instalados en una computadora de escritorio (como procesadores de palabras y programas de diseño, por ejemplo).

Por otro lado, tampoco debemos confundir las apps con las aplicaciones web, que son programas informáticos disponibles para su ejecución a través de la conexión con un servidor mediante internet, es decir, software disponible para ejecutarse directamente a través de un explorador web. (Quiroz, 2025)

### **3.11 Metodología de desarrollo de Software**

Las metodologías de desarrollo de software son un conjunto de técnicas y métodos que se aplican para diseñar soluciones de software informático. El objetivo de las distintas metodologías es el de intentar organizar los equipos de trabajo para que estos desarrollen las funciones de un programa de la mejor manera posible. (Open Academy, 2020)

### **3.12 Metodología ágil Scrum**

Scrum es un marco de trabajo ágil mediante el cual se puede abordar problemas complejos, adaptativos y realizando la entrega de productos de forma eficiente obteniendo el mejor resultado posible de un proyecto. (Martins, 2025)

## 4. Problema

### 4.1 Planteamiento del problema.

En Colombia durante los últimos años se ha incrementado las olas de calor, dado los fenómenos climáticos como el Niño, este ha representado un riesgo para la salud de los trabajadores, particularmente en el sector de la construcción. Ya que estar expuestos a altas temperaturas de calor puede generar problemas de deshidratación, agotamiento, insolación, afectando el rendimiento de los trabajadores.

Según Infobae (2024), Colombia ha experimentado fuertes olas de calor en gran parte del territorio, en regiones como Magdalena, Atlántico y Cesar. Esta situación ha alertado a los organismos de salud pública y ambientales ya que el país no está preparado a situaciones tan prolongadas. Según el artículo estas condiciones pueden agravar las condiciones de salud, sobre todo aquellos que trabajan al aire libre como los de la construcción.

Con base a lo anterior, queda claro que en muchos lugares de trabajo no cuentan sistemas de monitoreo en tiempo real que informen sobre condiciones térmicas peligrosas y que emitan tempranas en entornos laborales expuestos al calor.

Como solución, este proyecto plantea desarrollar un sistema de monitoreo basado en un casco inteligente con sensores de temperatura de humedad, temperatura corporal, temperatura ambiente y geolocalización, vinculado a una aplicación móvil. Permitiendo identificar de forma oportuna los signos de riesgo, enviando alertas para facilitar la intervención al personal encargado.

La justificación del Proyecto se sustenta en mejorar las herramientas y condiciones que impactan la salud de los trabajadores que están expuestos al calor extremo.

Además, responde a la necesidad de instituciones de salud y seguridad ocupacional que fomentan el uso de herramientas tecnológicas de prevención para minimizar accidentes laborales y enfermedades causadas por las oleadas de calor.

## 4.2 Pregunta de Investigación

¿Cómo ha afectado el cambio climático a los trabajadores del sector de la construcción según investigaciones realizadas?

## 5. Objetivos

### 5.1 Objetivo General

Desarrollar un sistema de monitoreo basado en un casco inteligente, equipado con sensores de temperatura, humedad y geolocalización, integrado a una aplicación móvil, que permita la recolección, procesamiento y visualización en tiempo real de condiciones térmicas del entorno laboral, con el fin de generar alertas preventivas que contribuyan a la protección de la salud y seguridad de los trabajadores del sector de la construcción.

### 5.2 Objetivos específicos

- Recolectar los requerimientos funcionales y no funcionales del sistema de monitoreo del casco inteligente, mediante el uso de técnicas de análisis de software como historias de usuario, encuestas e investigación del entorno de trabajo.
- Diseñar la arquitectura del sistema, incluyendo el diseño del casco inteligente con sus componentes electrónicos como sensores de temperatura, humedad y GPS también la estructura de la aplicación móvil, asegurando la integración entre hardware y software.
- Elaborar los diagramas de casos de uso, diagramas de clases, diagramas de flujo que representen el comportamiento lógico y la estructura del sistema, en función de los perfiles de usuario definidos: trabajador, SST y administrador.
- Desarrollar el prototipo del casco inteligente integrando el microcontrolador, los sensores, el módulo de GPS, junto con la aplicación móvil que recibe, procesa y muestra la información en tiempo real.
- Realizar pruebas funcionales, de integración y validación del sistema en escenarios reales, con el fin de evaluar el desempeño del casco inteligente y la aplicación

móvil, asegurando la confiabilidad, precisión de los datos y efectividad de las alertas generadas.

- Elaborar el manual de usuario y técnico, con el fin de brindar una herramienta de apoyo, los cuales facilitarán su uso, configuración y mantenimiento de la aplicación móvil, por parte de los diferentes perfiles involucrados.

## 6. Metodología

Para desarrollo del proyecto nos basamos en la metodología ágil Scrum.

La cual nos facilita gestionar de forma eficiente, el análisis, desempeño y logros proyectados. Scrum permite organizar de manera iterativa e incremental el trabajo, ya que se enfoca en entregar de manera temprana y oportuna valor al usuario. Este método resulta beneficioso para la integración tanto del hardware y software, garantizando la trazabilidad y flexibilidad ante cambios o mejoras propuestas por los usuarios.

### 6.1. Fases análisis

LA CUAL CORRESPONDE AL SIGUIENTE OBJETIVO

Recolectar los requerimientos funcionales y no funcionales del sistema de monitoreo del casco inteligente, mediante el uso de técnicas de análisis de software como historias de usuario, encuestas e investigación del entorno de trabajo.

En esta fase de análisis del proyecto Casco Inteligente, se halló un problema que afecta a los trabajadores de la industria de la construcción: su larga exposición a temperaturas de calor extremas. Estar expuestos puede causar problemas de agotamiento, deshidratación y golpes de calor comprometiendo la salud, rendimiento y seguridad laboral de los trabajadores.

Como solución inicial, se planeó la macro idea de elaborar un sistema de monitoreo bajo un casco inteligente que mida la temperatura en tiempo real, envié alertas al área de seguridad y salud en el trabajo. Surge la idea después de analizar las necesidades en el entorno laboral de la construcción y esta es apoyada por medio de investigaciones como el artículo de *Infobae, 2024* sobre la ola de calor en Colombia.

En esta fase se elaboró y aplicó una encuesta directamente a los trabajadores del sector de construcción, con el propósito de conocer su perspectiva con el uso de la tecnología en el ámbito laboral para prevenir riesgos provocados por las oleadas de calor. A partir de los resultados se consideraron las funcionalidades más estimadas por los usuarios, lo que permitió confirmar que el proyecto es viable y determinar un diseño y desarrollo del prototipo.

## 6.2 Formato de la encuesta:

1. ¿Le gustaría tener como apoyo a su labor un casco inteligente que le permita medir la temperatura y humedad para proteger su salud?

- Sí
- No
- No estoy seguro/a

2. ¿Cree que la implementación de este sistema puede reducir los accidentes laborales relacionados con el calor extremo?

- Sí
- No
- No estoy seguro/a

3. ¿Qué características del casco inteligente considera más importantes? (Seleccione las que apliquen)

- Medición de temperatura corporal
- Monitoreo de la humedad ambiental
- Alertas tempranas de golpe de calor
- Comodidad del casco
- Conectividad con dispositivos móviles
- Duración de la batería
- Facilidad de uso

4. ¿Le genera seguridad usar tecnología en su lugar de trabajo para monitorear su salud?

- Muy confiado/a
- Algo confiado/a
- No muy confiado/a
- Nada confiado/a

5. En cuanto a la calidad del casco (material, peso, etc.), ¿qué características considera más importantes?

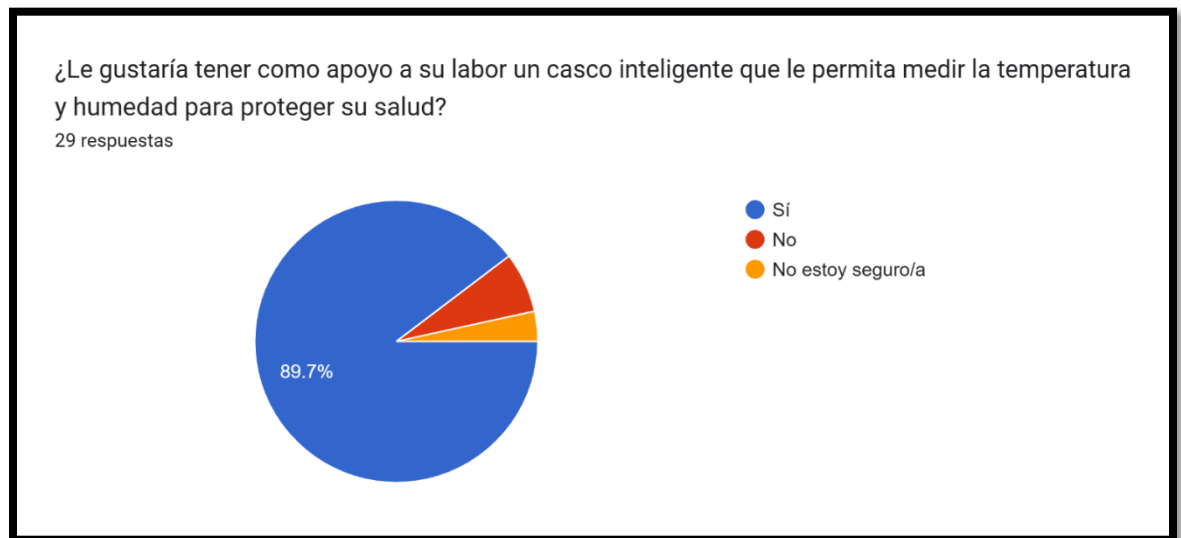
- Ligereza y comodidad
- Resistencia y durabilidad

FUENTE:

<https://docs.google.com/forms/d/1wN3ii99S3I80jehXBuPe2q8LCLsUDTAq3RFFN3t2jkY/edit>

### 6.3 Análisis de la encuesta:

- El 89.7% de los trabajadores encuestados ven a favor el uso del casco, lo cual arroja una viabilidad del proyecto.

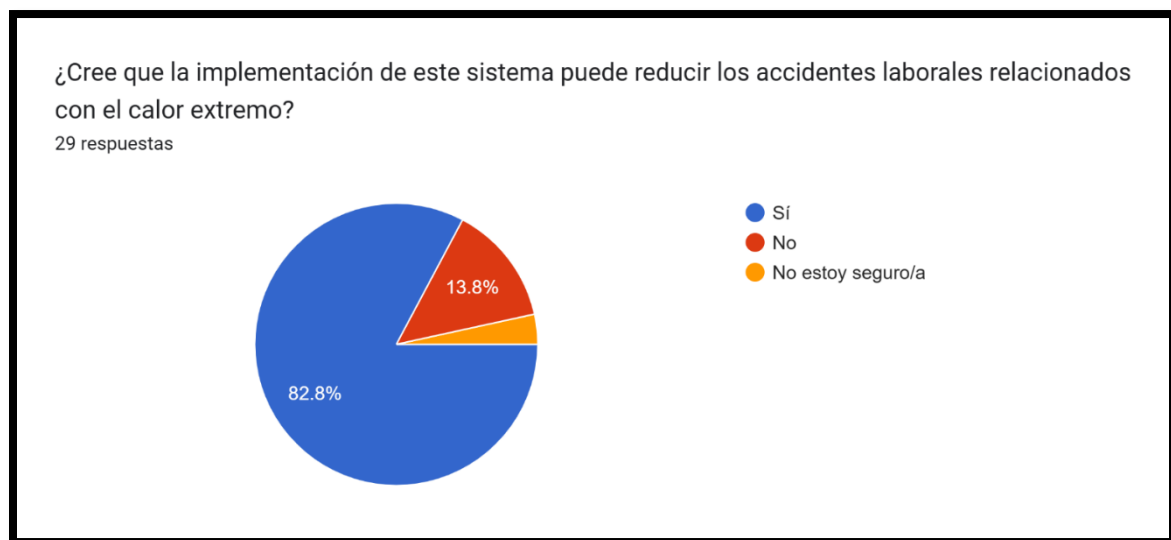


- **Figura 1 Resultados de la Respuesta**

FUENTE:

[HTTPS://DOCS.GOOGLE.COM/FORMS/D/1wN3ii99S3I80JEHXBuPe2q8LCLsUDTAq3RFFN3t2jkY/EDIT#RESPONSES](https://docs.google.com/forms/d/1wN3ii99S3I80jehXBuPe2q8LCLsUDTAq3RFFN3t2jkY/edit#RESPONSES)

- El 82.8% de los encuestados piensan que puede prevenir accidentes laborales, lo cual demuestra la orientación preventiva del proyecto.

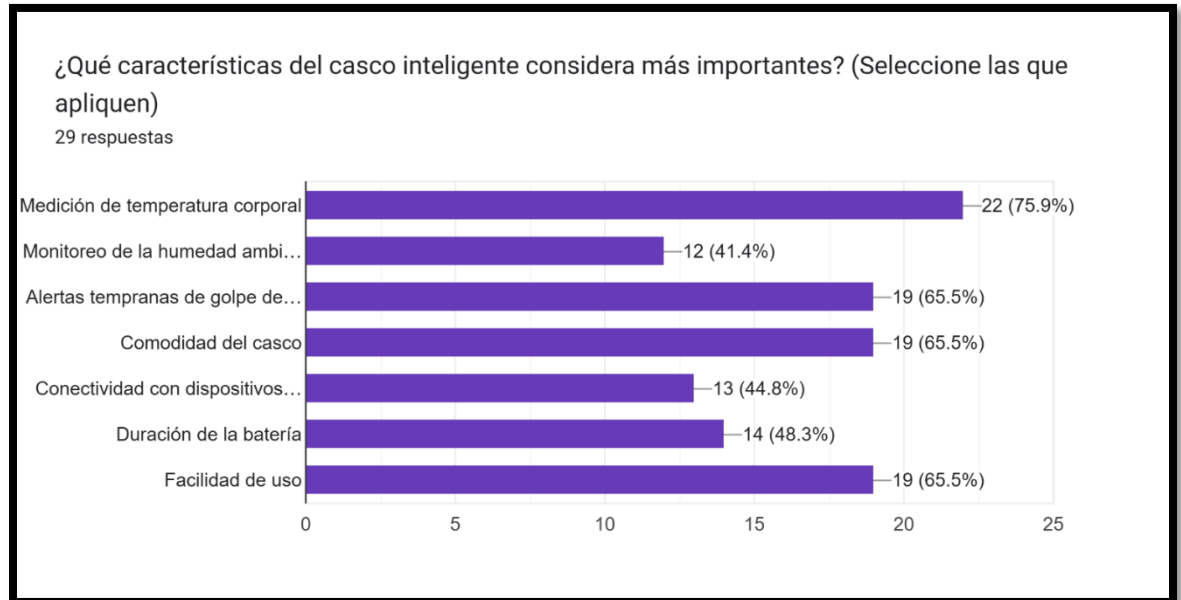


- **Figura 2 Resultados de la Respuesta**

FUENTE:

[HTTPS://DOCS.GOOGLE.COM/FORMS/D/1wN3iI99S3i80JEHXBuPe2q8LCLsUDTAQ3RFFN3T2JKY/EDIT#RESPONSES](https://docs.google.com/forms/d/1wN3iI99S3i80JEHXBuPe2q8LCLsUDTAQ3RFFN3T2JKY/edit#responses)

- Las funciones más apreciadas son: medición de temperatura corporal, alertas tempranas y comodidad, lo que orientó el diseño del mínimo producto viable.

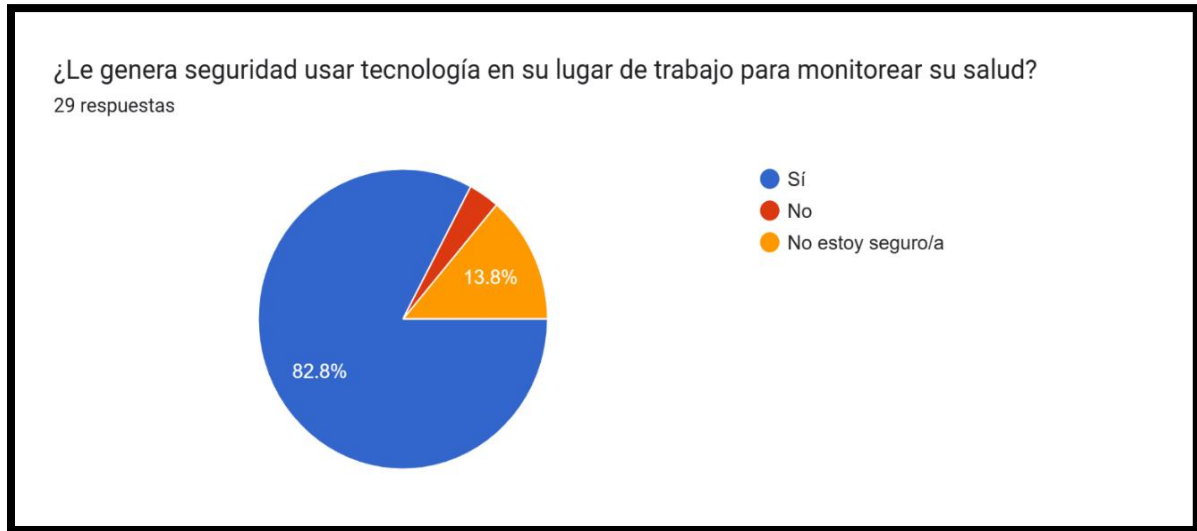


- **Figura 3 Resultados de la Respuesta**

FUENTE:

[HTTPS://DOCS.GOOGLE.COM/FORMS/D/1wN3ii99S3i80JEHXBuPe2q8LCLsUDTAQ3RFFN3T2JKY/EDIT#RESPONSES](https://docs.google.com/forms/d/1wN3ii99S3i80JEHXBuPe2q8LCLsUDTAQ3RFFN3T2JKY/edit#responses)

- El 83% de los trabajadores se sienten seguros haciendo uso de la tecnología para monitorear su salud, lo cual demuestra una alta adopción del casco inteligente en obras de construcción.



- **Figura 4 Resultados de la Respuesta**

FUENTE:

[HTTPS://DOCS.GOOGLE.COM/FORMS/D/1wN3ii99S3i80JEHXBuPe2Q8LCLsUDTAQ3RFFN3t2JKY/EDIT#RESPONSES](https://docs.google.com/forms/d/1wN3ii99S3i80JEHXBuPe2Q8LCLsUDTAQ3RFFN3t2JKY/EDIT#RESPONSES)

## 7. Conclusión de la encuesta

Los resultados de la encuesta muestran una buena aceptación del casco inteligente entre los trabajadores de construcción. La mayoría considera que esta herramienta puede ayudar a prevenir riesgos por calor extremo y reducir accidentes laborales, lo cual valida la pertinencia del proyecto.

Esto demuestra que existe una necesidad real en el sector y que los trabajadores están dispuestos a adoptar este tipo de tecnología, por lo que se puede continuar con mayor confianza en las etapas de diseño y desarrollo del sistema.

## 8. Historias de usuarios

- Como trabajador en el area de la construcción, quiero que el casco mida en tiempo real, la temperatura ambiente y la humedad, con el fin de recibir alertas anticipadas, cuando las condiciones sean peligrosas y poder tomar medidas preventivas.
- Como trabajador en el sector de la construcción, quiero que el casco registre mi temperatura corporal y ritmo cardíaco, para determinar si mi salud está en riesgo y poder tomar descansos adecuados.
- Como SST encargado de la obra, quiero recibir notificaciones en mi aplicación móvil cuando un trabajador esté en riesgo, para poder actuar de manera rapida y garantizar su seguridad.
- Como personal de SST, quiero tener acceso a un historial de condiciones ambientales y alertas de riesgo, para proceder al analisis de patrones y mejorar las condiciones laborales.
- Como SST, quiero que el sisistema sugiera medidas preventivas basadas en los datos recopilados, con el fin de implementar estrategias que disminuyan la exposision de condiciones de riesgo a los trabajadores.
- Como SST, quiero que el sistema registre incidentes y medidas correctivas tomadas, para dar seguimiento a cada caso y mejorar las políticas de seguridad.

- Como administrador del sistema, quiero gestionar usuarios y permisos de la plataforma, para asegurar que solo las personas autorizadas puedan acceder a información crítica.
- Como administrador del sistema, quiero configurar los límites de temperatura y humedad, para ajustar las alerta según las condiciones de cada obra.

### **8.1 Historias de Usuario Prioritarias (MVP - Versión 1.0)**

- Como trabajador en el area de la construcción, quiero recibir alertas inmediatas en mi casco cuando la temperatura y humedad sean peligrosas, para poder implementar medidas de seguridad de manera oportuna. Justificación: Es la funcionalidad es fundamental en el sistema, ya que sin alertas el casco no cumple su finalidad.
- Como SST encargado de la obra, quiero recibir notificaciones en mi aplicación móvil cuando un trabajador se encuentre en situacion de riesgo, para intervenir de manera rápida y prevenir incidentes. Justificación: Es necesario que alguien más tenga visibilidad del riesgo para asegurar una respuesta ágil.
- Como responsable de seguridad laboral, quiero acceder a un panel de monitoreo en tiempo real con datos de temperatura y alertas activadas, para supervisar las condiciones de los trabajadores en el sitio de construccion. Justificación: esto Permite un control centralizado y la posibilidad de prevenir riesgos a gran escala.
- Como sistema de sensores del casco, quiero medir en tiempo real la temperatura y humedad del ambiente del entorno, para identificar riesgos y activar alertas cuando los valores superen umbrales críticos. Justificación: Es la base del funcionamiento del casco, ya que sin sensores el sistema carece de datos para operar.

## 8.2 Historias de usuarios ajustadas

- Como trabajador en el area de la construcción, quiero que el casco me notifique mediante una alerta cuando la temperatura o humedad superen niveles peligrosos, para poder buscar sombra o tomar un descanso oportunamente.
- Como trabajador de construcción, quiero que el casco monitoree mi temperatura corporal y ritmo cardíaco, para detectar indicios de agotamiento por calor y salvaguardar mi salud durante la jornada laboral
- Como SST encargado de la obra, quiero recibir alertas en mi celular cuando un trabajador muestre signos de riesgo por calor, de manera que pueda asistirlo o implementar protocolos de emergencia de inmediato.
- Como SST encargado de la obra, requiero ver un historial de alertas y datos ambientales, para identificar qué áreas o en que momentos del día son más riesgosos y así prevenir incidentes futuros.
- Como SST, quiero consultar reportes qque contengan datos de calor extremo y alertas, con el propositio de analizar tendencias y tomar decisiones preventivas.
- Como SST, deseo ser notificado si en una obra ocurren alertas repetidas por calor, para investigar las causas y fortalecer las medidas de seguridad.
- Como SST, quiero que el sistema ssugiera acciones preventivas basadas en los datos, con el el objetivo de disminuir el riesgo de exposición a oleadas de calr fuertes.
- Como administrador del sistema, deseo tener control sobre quien puede acceder a cada módulo del sistema, con el fin de proteger la información sensible y garantizar la seguridad de los datos.
- Como administrador del sistema, quiero ajustar los niveles máximos permitidos de temperatura y humedad según el proyecto o ciudad, para que las alertas sean más precisas.

### 9. Alcance del sistema propuesto en términos de (entradas, procesos y salidas)

Entradas	Procesos	Salidas
Usuario	Gestión de perfiles	Informe de usuarios registrados
Temperatura ambiente y humedad	Lectura en tiempo real desde los sensores establecidos	Informe del estado de la temperatura de los usuarios
Temperatura corporal	Captura de la ubicación desde el módulo GPS	Consultar la ubicación en panel de monitoreo
Registro de tips	Registro automático cada vez que se detecta una condición crítica	Consultar Historial de alertas
Ubicación obra		Consultar los tips en el banner de la app
Asignar casco		Informe de activación de alertas críticas
Definir rangos de temperatura		

- **Tabla 2 Alcance del sistema**

FUENTE: ELABORACIÓN PROPIA.

## 10. Alcance del Sistema

Este sistema está diseñado para cuidar y proteger la salud de aquellos trabajadores en especial del área de construcción los cuales trabajan bajo condiciones de altas temperaturas debido a que constantemente están directamente expuestos bajo el sol. El prototipo facilitara la medición de la temperatura corporal, la temperatura ambiente y la humedad como factores más importantes, también se integrará un módulo GPS para al momento de generar una alerta critica esta vaya con la ubicación actual de ese empleado, la alerta se ara visible en tiempo real al área de seguridad y salud en el trabajo la cual mediante un módulo podrá monitorear en tiempo real todos los trabajadores activos y su estado de temperatura.

En esta fase, del proyecto se centra en un prototipo funcional que contribuya a prevenir riesgos relacionados con los golpes de calor. Durante esta etapa, no se contempla la integración con plataformas empresariales complejas, almacenamiento de datos en la nube o monitoreo médico avanzado. No obstante, el sistema presenta un punto de partida sólido que permitirá, en el futuro, avanzar hacia soluciones más integrales orientadas a la seguridad laboral.

## 11. Nombre de la aplicación

**CLIMA SEGURO:** Este nombre fue elegido porque representa la esencia del proyecto: un entorno de trabajo donde las condiciones climáticas son monitoreadas constantemente, para que contribuya a reducir los riesgos laborales causados por las condiciones climáticas.

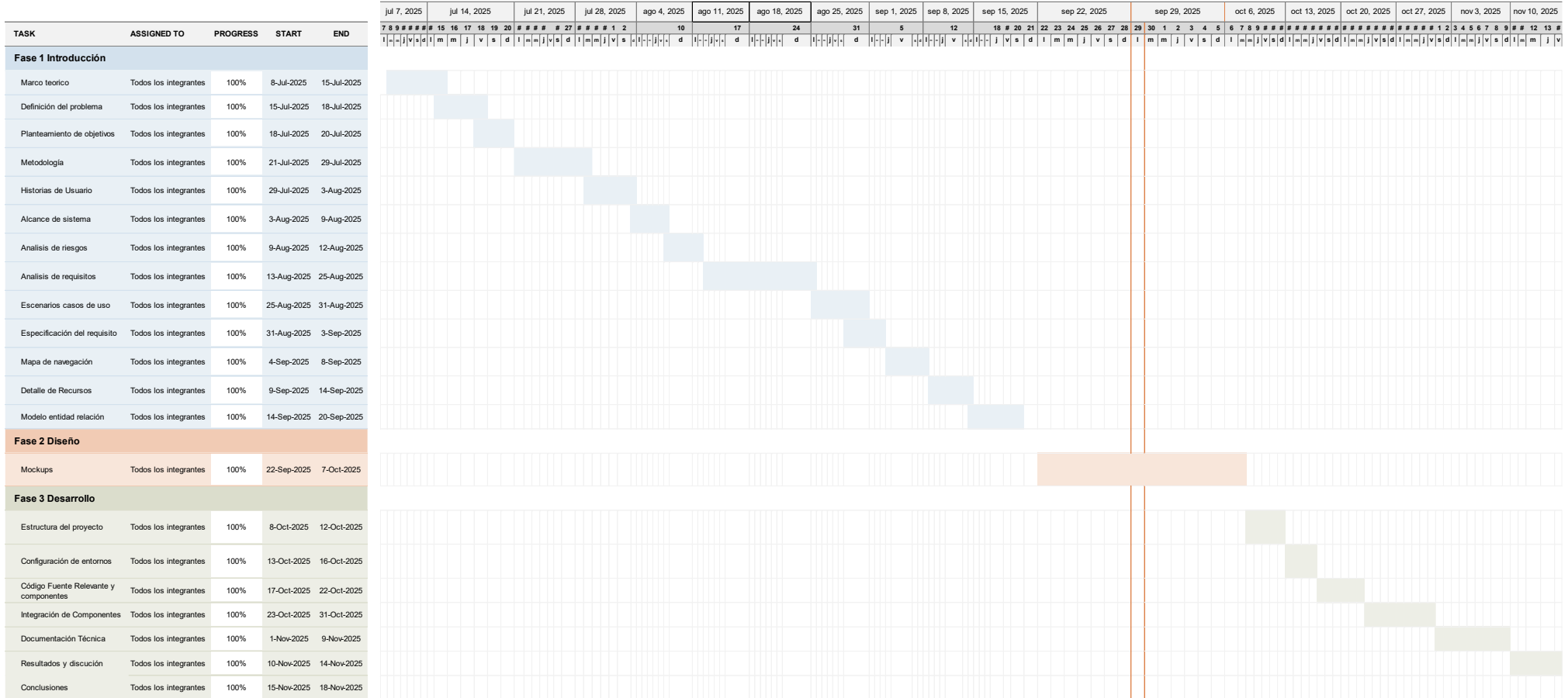
*Clima* hace referencia a los factores ambientales como la temperatura, la humedad y la exposición al calor extremo. *Seguro* representa el objetivo del sistema: generar alertas preventivas y facilitar la toma de decisiones oportunas para evitar incidentes relacionados con golpes de calor u otros efectos térmicos adversos.

## TRABAJO DE GRADO Opción Investigación o Proyecto de Grado

### 12. Cronograma de Actividades

Clima Seguro

Proyec **8-Jul-2025**  
Display **1**



**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**  
**13. Análisis de riesgos**

“El enfoque iterativo facilita la implementación de nuevos cambios tácticos de requerimientos y características del cronograma. Con este enfoque se fortalece tempranamente la identificación de los riesgos del proyecto, cuando aún es posible atacarlos y reaccionar a tiempo y eficientemente.” (Fernandez, 2013)

Según Javier Vargas los riesgos que se podrían llegar a gestionar en un producto son:

“Riesgo derivado del VALOR (¿la gente lo comprará o decidirá usarlo?).

Riesgo derivado de la USABILIDAD (¿pueden los usuarios descubrir cómo usarlo?).

Riesgo derivado de la FACTIBILIDAD (¿podemos construirlo con el tiempo, las habilidades y la tecnología que tenemos?).

Riesgo derivado de la VIABILIDAD de negocio (¿será viable económicamente? ¿ROI?).”

(Garzas, 2022)

Descripción del riesgo	Tipo de riesgo	Probabilidad de que ocurra	Plan de acción
Interrupciones del servicio	<b>Tecnológico</b>	Media	Contar con una infraestructura robusta y solida de servidores y Utiliza servicios en la nube confiables y considera la redundancia y la tolerancia a fallos para garantizar la disponibilidad continua de la aplicación.
Fallos en la aplicación		Media	Implementar sistemas de monitoreo en tiempo real para detectar posibles problemas de rendimiento
Falta de capacidad para funcionamiento de la herramienta		Media	Realiza pruebas exhaustivas de carga y rendimiento en la aplicación para identificar posibles cuellos de botella y asegurarte de que la infraestructura pueda manejar picos de tráfico sin problemas

Violación de seguridad		Media	Mantener el antivirus actualizado y correcto mantenimiento para garantizar su seguridad.
Incidentes mientras se utiliza la aplicación		Alto	Establecer un equipo de respuesta ante incidentes que pueda actuar rápidamente para solucionar problemas técnicos y restaurar la funcionalidad de la aplicación en caso de interrupciones del servicio
Falla en la estabilidad de la aplicación		Media	Mantener la aplicación actualizada y realizar mantenimiento regular en la infraestructura para garantizar su estabilidad.
Daño del equipo		Bajo	Tener un equipo de repuesto o a la mano que pueda utilizarse en caso de daño.
Alcance mal definido	<b>Proceso</b>	Bajo	Definir los requisitos y objetivos del proyecto desde el inicio.
Cambios frecuentes en los requisitos		Medio	
Falta de planificación y seguimiento		Bajo	Planificación adecuada y un seguimiento riguroso del progreso del proyecto.
Falta de pruebas adecuadas		Medio	Realizar las pruebas que sean necesarias para garantizar el correcto funcionamiento de la aplicación.

Dependencia excesiva de tecnologías emergentes		Bajo	No adoptar tecnologías nuevas y no probadas sin comprender completamente sus implicaciones.
Falta de atención a la experiencia del usuario		Bajo	Tener en cuenta los principios de diseño centrado en el usuario, para no afectar su adopción y satisfacción con la aplicación.
Que el cliente este insatisfecho con el producto final	<i>Cliente</i>	Bajo	Trabaja estrechamente con el cliente para definir claramente los requisitos del proyecto y establecer expectativas realistas desde el principio.
Requisitos ambiguos o incompletos		Medio	
Falta de participación o comunicación con el administrador		Bajo	Mantener una comunicación abierta y transparente con el cliente durante todo el proceso de desarrollo.
Que el cliente no sepa interactuar con la aplicación		Medio	Proporcionar documentación detallada que explique cómo utilizar la herramienta paso a paso, incluyendo guías de usuario, tutoriales y videos explicativos
Poca aceptación de la plataforma		Bajo	Ofrecer sesiones de capacitación personalizadas para el cliente, adaptadas a sus necesidades y nivel de experiencia.

- **Tabla 3 Análisis de Riesgos**

FUENTE: ELABORACIÓN PROPIA.

## 14. Análisis de Requisitos

“En esta parte el ingeniero intenta comprender la naturaleza de los programas que han de construirse, así como el dominio de la aplicación.

Diseño: En esta fase se traducen los requisitos a una representación que pueda ser evaluada previamente antes de empezar la fase de codificación.” (Fernandez, 2013)

### 14.1 Actores del sistema

Actor	Función
Trabajador	Es la persona la cual portara el casco inteligente, podra visualizar su estado de temperatura y la temperatura ambiente, recibir alertas en tiempo real y de igual manera podra ver tips preventivos frente a un posible riesgo de golpe de calor
SST	Es el encargo de monitorear el sistema en tiempo real, Visualizar alertas generadas por el sistema cuando se superen los rangos térmicos establecidos y tomar decisiones frente a situaciones de emergencia.
Administrador del sistema	la persona encargada de gestionar el mantenimiento del aplicativo, registrar y gestionar usuarios, visualizar reportes de emergencias registradas, controlar permisos y gestionar los tips de emergencia.

- **Tabla 4 Actores del Sistema**

FUENTE: ELABORACIÓN PROPIA.

### 14.2 Tabla general para casos de uso

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Trabajador	Crear	x	x	x
	Consultar	x	x	x
	Modificar			x
	Guardar	x	x	x
	Inhabilitar			x
	Cancelar	x	x	x
	Salir	x	x	x

- **Tabla 5 general para casos de uso**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Temperatura ambiente humedad	Crear			x
	Consultar	x	x	x
	Modificar			x
	Guardar			x
	Inhabilitar			x
	Cancelar	x	x	x

	Salir	x	x	x
--	-------	---	---	---

- **Tabla 6 Temperatura ambiente y humedad**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Temperatura corporal	Crear			x
	Consultar	x	x	x
	Modificar			x
	Guardar			x
	Inhabilitar			x
	Cancelar	x	x	x
	Salir	x	x	x

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Registro de tips	Crear		x	x
	Consultar	x	x	x
	Modificar		x	x
	Guardar		x	x
	Inhabilitar			x
	Cancelar	x	x	x

	Salir	x	x	x
--	-------	---	---	---

- **Tabla 7 Temperatura corporal y Registro de tips**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Ubicación de obra	Crear			x
	Consultar		x	x
	Modificar			x
	Guardar			x
	Inhabilitar			x
	Cancelar		x	x
	Salir		x	x

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Asignar casco	Crear			x
	Consultar	x	x	x
	Modificar			x
	Guardar			x
	Inhabilitar			x
	Cancelar	x	x	x
	Salir	x	x	x

- **Tabla 8 Ubicación de obra y Asignar casco**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Definir rangos de temperatura	Crear			x
	Consultar		x	x
	Modificar			x
	Guardar			x
	Inhabilitar			x
	Cancelar			x
	Salir		x	x

- **Tabla 9 Definir rangos de temperatura**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Perfil de Usuario	Crear	x	x	x
	Consultar	x	x	x
	Modificar	x	x	x
	Guardar	x	x	x
	Inhabilitar			x
	Cancelar	x	x	x

	Salir	x	x	x
--	-------	---	---	---

- **Tabla 10 Perfil de Usuario**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Usuario	Crear	x	x	x
	Consultar	x	x	x
	Modificar	x	x	x
	Guardar	x	x	x
	Inhabilitar			x
	Cancelar	x	x	x
	Salir	x	x	x

- **Tabla 11 Usuario**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
	Consultar la ubicacion en panel de monitoreo		x	x
	Consultar historial de alertas		x	x

Gestión consulta	Consultar los tips en el banner de la app	x	x	x
------------------	---	---	---	---

- **Tabla 12 Gestión Consulta**

Gestiones	Acciones	Actores		
		Usuario	SST	Administrador del sistema
Gestión informes	Informe de usuarios registrados		x	x
	Informe del estado de la temperatura de los usuarios		x	x
	Informe de activación de alertas críticas		x	x

- **Tabla 13 Gestión Informes**

FUENTE: ELABORACIÓN PROPIA.

### 14.3 Requisitos de usuario (RU)

Los requisitos de usuario describen lo que necesitan y esperan las personas que van a usar el sistema. En este proyecto, se identificaron las necesidades de tres perfiles: trabajadores de construcción, personal de SST y administradores del sistema.

Estos requisitos sirven como base para desarrollar un sistema que realmente responda a las necesidades del entorno laboral y sea fácil de usar para quienes lo operarán diariamente.

Está dirigida tanto al cliente como al equipo de desarrollo. El lenguaje utilizado para su redacción debe ser informal, de forma que sea fácilmente comprensible para todas las partes involucradas en el desarrollo. (Especificación de requisitos de software, 2024)

ID	Nombre	Descripción	Usuario
RU-001	Trabajador	El sistema permitirá la gestión de la información del trabajador para crear, consultar, guardar, cancelar, y salir.	Trabajador, SST y administrador.
RU-002	Temperatura ambiente y humedad	El sistema permitirá la gestión relacionada con la información de la temperatura ambiente y humedad para consultar, cancelar y salir.	Trabajador, SST y administrador.
RU-003	Temperatura corporal	El sistema permitirá la gestión relacionada con la información de la temperatura corporal para consultar, cancelar y salir.	Trabajador, SST y administrador.
RU-004	Registro de tips	El sistema permitirá la gestión de la información de los registros de tips para crear, consultar, guardar, cancelar, y salir.	SST y administrador.
RU-005	Ubicación de obra	El sistema permitirá la gestión de la información de ubicación de obra para crear, consultar, guardar, cancelar, y salir.	SST y administrador.
RU-006	Asignar casco	El sistema permitirá la gestión de la información de la asignación de cascos para crear, consultar, guardar, cancelar, y salir.	Trabajador, SST y administrador.
RU-07	Definir rangos de temperatura	El sistema permitirá la gestión de la configuración de definir los rangos de la temperatura crítico para crear, consultar, guardar, cancelar, y salir.	SST y administrador.
RU-08	Perfil de Usuario	El sistema permitirá la gestión de la información de registro de perfil de usuario para crear, consultar, cancelar, y salir.	Trabajador, SST y administrador.

RU-09	Usuario	El sistema permitirá la gestión de la información de usuario para crear, consultar, cancelar, y salir.	Trabajador, SST y administrador.
RU-10	Gestión consulta	El sistema permitirá la gestión de la información de consulta para crear, consultar, cancelar, y salir.	Trabajador, SST y administrador.
RU-11	Gestión Informes	El sistema permitirá la gestión de informes sobre la información registrada para crear, consultar, cancelar, y salir.	SST y administrador.

- **Tabla 14 Requisitos de Usuarios**

FUENTE: ELABORACIÓN PROPIA.

#### 14.4 Requisitos funcionales (RF)

“Expresan la naturaleza del funcionamiento del sistema (cómo interacciona el sistema con su entorno y cuáles van a ser su estado y funcionamiento).

Nota: A veces, también es conveniente indicar lo que no hará el sistema.”

ID RF	Nombre	Descripción	Usuario	ID RU
RF-001	Crear trabajador	Permite registrar la información del trabajador con los siguientes datos: Nombres, Apellidos, Cedula, Fecha de nacimiento, Teléfono, Dirección, Correo electrónico, Contraseña y Confirmar contraseña	Trabajador, SST y administrador	RU-001

RF-002	Consultar trabajador	Permite consultar la información del usuario registrado en la aplicación por el campo cédula.	Trabajador, SST y administrador	RU-001
RF-003	Modificar trabajador	Permite realizar modificaciones a los datos del usuario registrados en la aplicación a excepción del campo clave.	Administrador	RU-001
RF-004	Guardar trabajador	Permite guardar los cambios realizados en el usuario.	Trabajador, SST y administrador	RU-001
RF-005	Inhabilitar trabajador	Permite inhabilitar o habilitar el usuario para usar la aplicación con previa verificación.	Administrador	RU-001
RF-006	Cancelar trabajador	Permite cancelar la vista actual sin guardar los cambios realizados	Trabajador, SST y administrador	RU-001
RF-007	Salir trabajador	Permite salir del formulario de trabajador	Trabajador, SST y administrador	RU-001
RF-008	Crear temperatura ambiente y humedad	Permite registrar el rango de temperatura y humedad del ambiente	Administrador	RU-002
RF-009	Consultar temperatura ambiente y humedad	Permite consultar la temperatura actual	Trabajador, SST y administrador	RU-002

RF-010	Modificar temperatura ambiente y humedad	Permite modificar los datos de la temperatura y el ambiente	Administrador	RU-002
RF-011	Guardar temperatura ambiente y humedad	Permite guardar los datos de la temperatura que se capturan en la aplicación.	Administrador	RU-002
RF-012	Inhabilitar temperatura ambiente y humedad	Permite inhabilitar o habilitar la temperatura ambiente y humedad con previa verificación.	Administrador	RU-002
RF-013	Cancelar temperatura ambiente y humedad	Permite cancelar la vista actual sin guardar los cambios realizados	Trabajador, SST y administrador	RU-002
RF-014	Salir temperatura ambiente y humedad	Permite salir de la vista actual	Trabajador, SST y administrador	RU-002
RF-015	Crear temperatura corporal	Permite registrar desde la base de datos los rangos de temperatura corporal	Administrador	RU-003
RF-016	Consultar temperatura corporal	Permite consultar la información de la temperatura corporal capturada en la aplicación	Trabajador, SST y administrador	RU-003

RF-017	Modificar temperatura corporal	Permite modificar los datos de la temperatura corporal	Administrador.	RU-003
RF-018	Guardar temperatura corporal	Permite guardar los datos de la temperatura corporal	Administrador.	RU-003
RF-019	Inhabilitar temperatura corporal	Permite inhabilitar o habilitar la temperatura corporal con previa verificación.	Administrador.	RU-003
RF-020	Cancelar temperatura corporal	Permite cancelar la vista actual sin guardar los cambios realizados	Trabajador, SST y administrador	RU-003
RF-021	Salir temperatura corporal	Permite salir de la vista actual	Trabajador, SST y administrador	RU-003
RF-022	Crear tips de Seguridad	Permite crear registros de tips relacionados con la seguridad en el trabajo.	Administrador	RU-004
RF-023	Consultar tips de Seguridad	Permite realizar consultas sobre los diferentes tips de seguridad registrados	Trabajador, SST y administrador	RU-004
RF-024	Modificar tips de Seguridad	Permite realizar modificaciones sobre los tips existentes	SST, Administrador	RU-004
RF-025	Guardar tip de seguridad	Permite guardar la información de los	SST, Administrador	RU-004

		cambios realizados en los tips		
RF-026	Inhabilitar tip de seguridad	Permite inhabilitar o habilitar tips de seguridad	Administrador	RU-004
RF-027	Cancelar tips de Seguridad	Permite cancelar la vista actual sin guardar cambios	Trabajador, SST y administrador	RU-004
RF-028	Salir tips de seguridad	Permite salir de la vista de tips preventivos	Trabajador, SST y administrador	RU-004
RF-029	Crear Ubicación de obra	Permite crear los diferentes sitios de la obra	Administrador	RU-005
RF-030	Consultar Ubicación de obra	Permite realizar consultas sobre la información de la ubicación de obra	SST, Administrador	RU-005
RF-031	Modificar Ubicación de obra	Permite realizar modificaciones sobre la información de la ubicación de obra	Administrador	RU-005
RF-032	Guardar ubicación de obra	Permite guardar registros de la información de ubicación de obra.	Administrador	RU-005
RF-033	Inhabilitar Ubicación de obra	Permite inhabilitar o habilitar ubicación de obra con previa verificación.	Administrador	RU-005
RF-034	Cancelar Ubicación de obra	Permite cancelar la gestión de ubicación de obra	SST, Administrador	RU-005

		sin guardar los cambios realizados		
RF-035	Salir de ubicación de obra	Permite salir de la vista actual de ubicación de obra	SST, Administrador	RU-005
RF-036	Crear asignación de casco	Permite registrar la información de la asignación de casco a un usuario	Administrador	RU-006
RF-037	Consultar asignación de casco	Permite realizar consultas sobre la información de los cascos asignados	SST, Administrador	RU-006
RF-038	Modificar asignación de casco	Permite realizar modificaciones sobre la información de un casco asignado a un usuario	Administrador	RU-006
RF-039	Guardar asignación de casco	Permite guardar la información de un casco asignado	Administrador	RU-006
RF-040	Inhabilitar asignación de casco	Permite inhabilitar o habilitar la asignación de un casco existente	Administrador	RU-006
RF-041	Cancelar asignación de casco	Permite cancelar la asignación de un casco	Administrador	RU-006
RF-042	Salir de asignación de casco	Permite salir de la vista de asignación de casco sin guardar cambios	Administrador	RU-006
RF-043	Crear rangos de temperatura	Permite crear los rangos de temperatura críticos	Administrador	RU-007

RF-044	Consultar rangos de temperatura	Permite consultar los rangos de temperatura críticos	SST, Administrador	RU-007
RF-045	Modificar rangos de temperatura	Permite modificar los rangos de temperatura críticos	Administrador	RU-007
RF-046	Guardar rangos de temperatura	Permite guardar los rangos de temperatura críticos	Administrador	RU-007
RF-047	Inhabilitar rangos de temperatura	Permite inhabilitar o habilitar los rangos de temperatura críticos	Administrador	RU-007
RF-048	Cancelar rangos de temperatura	Permite cancelar los rangos de temperatura críticos	Administrador	RU-007
RF-049	Salir rangos de temperatura	Permite salir de la vista de asignación de los rangos de temperatura críticos	SST, Administrador	RU-007
RF-050	Crear perfil de usuario	Permite crear un perfil detallado para todos los actores de la aplicación con los siguientes datos: Código, Nombre.	Trabajador, SST y administrador.	RU-008
RF-051	Consultar perfil de usuario	Permite consultar el perfil de usuario existente por el campo código.	Trabajador, SST y administrador	RU-008
RF-052	Modificar perfil de usuario	Permite modificar los perfiles de	Trabajador, SST y	RU-008

		usuario existentes a excepción del campo clave.	administrador	
RF-053	Guardar perfil de usuario	Permite guardar la información del perfil de usuario que se ingrese en la aplicación.	Trabajador, SST y administrador	RU-008
RF-054	Inhabilitar perfil de usuario	Permite inhabilitar o habilitar los registros de perfil de usuario con previa verificación.	SST y administrador	RU-008
RF-055	Cancelar perfil de usuario	Permite cancelar la vista actual sin guardar los cambios realizados	Trabajador, SST y administrador	RU-008
RF-056	Salir perfil de usuario	Permite salir del formulario de perfil de usuario	Trabajador, SST y administrador	RU-008
RF-057	Crear usuario	Permite crear un usuario para todos los actores de la aplicación con los siguientes datos: Código, Usuario, Contraseña	Trabajador, SST y administrador	RU-009
RF-058	Consultar usuario	Permite consultar el usuario registrado en la aplicación por el campo código.	Trabajador, SST y administrador	RU-009
RF-059	Modificar usuario	Permite modificar el usuario registrado a excepción del campo clave.	Trabajador, SST y administrador	RU-009

RF-060	Guardar usuario	Permite guardar el usuario que se ingrese en la aplicación.	Trabajador, SST y administrador	RU-009
RF-061	Inhabilitar usuario	Permite inhabilitar o habilitar el usuario con previa verificación.	SST y administrador	RU-009
RF-062	Cancelar usuario	Permite cancelar la vista actual sin guardar los cambios realizados	Trabajador, SST y administrador	RU-009
RF-063	Salir usuario	Permite salir del formulario de usuario	Trabajador, SST y administrador	RU-009
RF-064	Crear gestión consulta	Permite crear consulta de la ubicación en panel de monitoreo, historial de alertas y los tips en el banner de app	SST y administrador	RU-010
RF-065	Gestión consulta	Permite realizar consultas sobre ubicación en panel de monitoreo, historial de alertas y los tips en el banner de app	SST y administrador	RU-010
RF-066	Modificar gestión consulta	Permite realizar modificaciones a las consultas a excepción del campo clave.	SST y administrador	RU-010
RF-067	Guardar gestión consulta	Permite guardar la consulta que se solicite en la aplicación.	SST y administrador	RU-010

RF-068	Inhabilitar gestión consulta	Permite inhabilitar o habilitar las consultas con previa verificación.	Administrador	RU-010
RF-069	Cancelar gestión consulta	Permite cancelar la vista actual sin guardar los cambios realizados	SST y administrador	RU-010
RF-070	Salir gestión consulta	Permite salir del formulario de gestión consulta	SST y administrador	RU-010
RF-071	Crear gestión informes	Permite crear gestión de informes tales como: usuarios registrados, Estado de temperatura de usuarios, Activación de alertas críticas	SST y administrador	RU-011
RF-072	Consultar gestión informes	Permite realizar consultas sobre la gestión de informes por los campos código y cédula.	SST y administrador	RU-011
RF-073	Modificar gestión informes	Permite realizar modificaciones a la gestión de informes a excepción del campo clave.	Administrador	RU-011
RF-074	Guardar gestión informes	Permite guardar la gestión de informes.	SST y administrador	RU-011
RF-075	Inhabilitar gestión informes	Permite inhabilitar o habilitar el ingreso de gestión	Administrador	RU-011

		de informes con previa verificación.		
RF-076	Cancelar gestiones informes	Permite cancelar la vista actual sin guardar los cambios realizados	SST y administrado r	RU-011
RF-077	Salir gestión informes	Permite salir del formulario de gestión de informes	SST y administrado r	RU-011

- **Tabla 15 Requisitos Funcionales**

FUENTE: ELABORACIÓN PROPIA.

## 14.5 Requisitos no funcionales

Restricciones sobre el espacio de posibles soluciones

Rendimiento del sistema: Fiabilidad, tiempo de respuesta, disponibilidad...

- Interfaces: Dispositivos de E/S, usabilidad, interoperabilidad...
- Proceso de desarrollo: Estándares, herramientas, plazo de entrega...

A los requisitos no funcionales se les suele llamar coloquialmente “cualidades” del sistema [“-ilities” en inglés] y pueden dividirse en dos categorías:

- Cualidades de ejecución, como la seguridad o la usabilidad, observables en tiempo de ejecución.
- Cualidades de evolución, como la “testabilidad”, mantenibilidad, extensibilidad o escalabilidad, determinadas por la estructura estática del software. (Granada, 2025)

**Tabla Facilidad de uso (“usability”)**

ID. Requisito	Descripción del requisito
RNF-001	Capacitación, antes de intensificar el uso del sistema los usuarios deben conocer su modo de uso.
RNF-002	Actualizar la información por ingreso o retiro de algún funcionario.
RNF-003	Diseño adecuado a las necesidades del usuario, para que la aplicación sea intuitiva y sencilla de usar cumpliendo con los siguientes parámetros: Tendrá una interfaz atrayente: formación de los elementos acorde al diseño. La carga de información deberá ser rápida.

- **Tabla 16 Requisitos No Funcionales**

Fuente: Elaboración Propia.

**Tabla Confiabilidad**

ID. Requisito	Descripción del requisito
RNF-001	El sistema debe estar disponible las 24 horas del día
RNF-002	Debe asegurar la permanente actualización de la base de datos, cuando se registre la información.

- **Tabla 17 Confiabilidad**

Fuente: Elaboración Propia.

**Tabla Ambiente de trabajo "Performance"**

ID. Requisito	Descripción del requisito
RNF-001	Tiempo de respuesta: se espera minimizar el tiempo a un promedio de 20 segundos, con el fin de que no se haga muy pesada la interacción con la Base de Datos manejada vía web.
RNF-002	Asignar suficiente espacio a la base de datos para soportar las grandes cantidades de información suministradas.
RNF-003	Asignar suficiente espacio a la base de datos para soportar las grandes cantidades de información suministradas.

- **Tabla 18 Performance**

Fuente: Elaboración Propia.

**Tabla Restricciones de diseño**

ID. Requisito	Descripción del requisito
RNF-001	El lenguaje de programación del sistema se espera implementar en Dart y su framework Flutter.
RNF-002	Se requiere de licenciamiento para el desarrollo del software.

- **Tabla 19 Restricciones de diseño**

Fuente: Elaboración Propia.

**Tabla Seguridad**

ID. Requisito	Descripción del requisito
RNF-001	Encriptación de las claves.
RNF-002	Realizar copia de seguridad, automático cada 6 horas, en servidor ubicado en otro espacio fuera de la empresa.
RNF-003	Los usuarios deberán estar registrados, bajo la modalidad del perfil del sistema.

- **Tabla 20 Seguridad**

Fuente: Elaboración Propia.

**Tabla Documentación de usuario y sistemas de ayuda**

ID. Requisito	Descripción del requisito
RNF-001	Capacitación a los usuarios del sistema, con el fin de lograr un buen manejo del mismo.
RNF-002	Capacitación a los usuarios del sistema, con el fin de lograr un buen manejo del mismo.

- **Tabla 21 Documentación de usuario y sistemas de ayuda**

Fuente: Elaboración Propia.

**Tabla Interfaz de Usuario**

ID. Requisito	Descripción del requisito
RNF-001	Será manejado a través de un computador de escritorio o dispositivo móvil.
RNF-002	Los colores de la aplicación serán gris oscuro, azul claro, blanco y negro.
RNF-003	El texto será manejado alternandose entre color negro y blanco, la fuente de la letra será Arial tamaño 12.

- **Tabla 22 Interfaz de Usuario**

Fuente: Elaboración Propia.

**Tabla Interfaces de comunicación**

ID. Requisito	Descripción del requisito
RNF-001	El acceso al software será vía web, debe asegurarse la estabilidad y seguridad de la conexión.
RNF-002	Comunicación con la interfaz de usuario.

- **Tabla 23 Interfaces de comunicación**

FUENTE: ELABORACIÓN PROPIA.

### 15. Escenarios de los Casos de Uso

Son muy útiles para explicar el funcionamiento del sistema, priorizar requerimientos cuando el sistema se desarrolla de forma incremental, elaborar manuales de usuario y especificar pruebas de aceptación.

- Describen el modo en que un actor interactúa con el sistema (descripción de un rol en lenguaje natural).
- Narran el comportamiento dinámico del sistema desde un punto de vista concreto (el del actor).
- Pueden expresar tanto requerimientos funcionales como no funcionales.
- Mejoran la trazabilidad de los requerimientos durante el proceso de desarrollo de software.
- Se pueden desarrollar en paralelo con los requerimientos del sistema de forma iterativa. (Granada, 2025)

Nombre		Crear Trabajador
Descripción		Permite registrar la información del usuario con los siguientes datos: Nombres, Apellidos, Cedula, Fecha de nacimiento, Teléfono, Dirección, Correo electrónico, Contraseña y Confirmar contraseña.
Actor		Trabajador, SST y administrador
Precondiciones		El usuario debe estar logueado en el sistema
Flujo básico		
	Actor	Sistema
	1. El usuario ingresa al menú Gestión / Trabajador.	2. El sistema despliega la interfaz con las

Pasos		opciones: Nombres, Apellidos, Cedula, Fecha de nacimiento, Teléfono, Dirección, Correo electrónico, Contraseña y Confirmar contraseña
	3. El usuario da clic en botón registrar.	4. El sistema activa todos los campos y envía el cursor al campo "ingrese sus nombres"
	5. El usuario ingresa la cédula.	El sistema verifica si la cédula existe.
	7. El usuario ingresa su fecha de nacimiento.	El sistema valida si es mayor de 18 años
	6. El usuario ingresa los demás datos del registro y va dando enter para llenarlos todos.	7. El sistema valida los campos requeridos
	8. El actor da clic en registrar.	9. El sistema almacena y arroja un mensaje de

		transacción exitosa.
	10. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		10. El sistema confirma que hay inconsistencia en los datos y arroja el mensaje de alerta
	11. El usuario da click en el botón cancelar	12. El sistema ubica el cursor en el campo "ingrese sus nombres"
	13. El usuario ingresa los datos correctamente y da clic en guardar.	14. El sistema verifica los datos nuevamente.
		15. El sistema almacena y arroja un mensaje de transacción exitosa.
	16. Clic en salir	
PostCondiciones	Existe un nuevo trabajador Registrado.	
Requisito Funcional	RF-001	

- **Tabla 24 ESCENARIOS DE LOS CASOS DE USO**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Consultar Trabajador</b>
Descripción		Permite consultar la información de los usuarios registrados en la aplicación
Actor		Trabajador, SST y administrador.
Precondiciones		El usuario debe estar logueado en el sistema
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión / Trabajador	2. El sistema despliega la interfaz con la visual de algunos empleados registrados recientemente, una caja de búsqueda la cual filtra por nombre o cedula.
	3. El usuario ingresa los parámetros de búsqueda y da clic en el botón Consultar.	4. El sistema procesa la información y

		muestra los usuarios que coincidan con los datos
	5. El actor selecciona un usuario de la lista	6. El sistema muestra la información detallada del usuario seleccionado
	8. Clic en salir	
<b>Flujo alternativo</b>		
	Actor	Sistema
Pasos	10. El actor ingresa los datos y da clic en consultar	9. El sistema procesa la información de los datos ingresados
	10. El actor ingreso datos inexistentes o incorrectos	11. El sistema muestra un mensaje: "Usuario no encontrado"
	13. Clic en Salir	16. Clic en salir
PostCondiciones	Se realizó una consulta exitosa de usuario registrado.	

Requisito Funcional	RF-002	
---------------------	--------	--

- **Tabla 25 Consultar Trabajador**

Fuente: Elaboración propia

Nombre		Modificar Trabajador
Descripción		Permite modificar o bien sea actualizar los datos de un usuario registrado en el sistema.
Actor		SST y administrador
Precondiciones		El usuario debe estar logueado en el sistema
Flujo básico		
Pasos	Actor	Sistema
	1. El actor ingresa al menú Gestión / trabajador	2. El sistema despliega la interfaz con una lista de usuarios registrados
	3. El actor selecciona un usuario y da clic en "editar"	4. El sistema activa los campos del formulario con la información previa

		registrada por el usuario
	5. El actor modifica los datos que requiera	El sistema verifica los nuevos datos ingresados
	7. El actor da clic en el botón "guardar"	El sistema actualiza la información y refresca la interfaz con los nuevos datos cargados
	10. Clic en salir.	
<b>Flujo alternativo</b>		
	Actor	Sistema
Pasos	11. El actor deja campos obligatorios vacios	10. El sistema indica con un mensaje de error los campos obligatorios vacios
	11. El actor completa los campos obligatorios	12. El sistema valida nuevamente y actualiza la información si todo está correcto
	16. Clic en salir	
PostCondiciones	Se modifiko un usuario exitosamente	

Requisito Funcional	RF-003	
---------------------	--------	--

- **Tabla 26 Modificar Trabajador**

Fuente: Elaboración propia

Nombre		Guardar Trabajador
Descripción		Permite guardar en la base de datos los usuarios nuevos registrados o las modificaciones realizadas a un usuario ya existente
Actor		Trabajador, SST, Administrador
Precondiciones		El usuario debe estar logueado en el sistema.
Flujo básico		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión/ Trabajador.	2. El sistema despliega la interfaz con las opciones: Nombres, Apellidos, Cedula, Fecha de nacimiento, Teléfono, Dirección, Correo electrónico, Contraseña y

		Confirmar contraseña
	2. El actor llena los datos del formulario y da clic en el botón "registrarse"	3. El sistema valida los datos ingresados y muestra un mensaje de Usuario registrado exitosamente
	3. El actor modifica los datos de un usuario existente en base de datos da clic en el botón "guardar"	4.El sistema verifica la información actualizada y refresca la pagina
	14. Clic en salir.	
<b>Flujo alternativo</b>		
	Actor	Sistema
Pasos	El actor omite algún campo obligatorio o ingresa un formato incorrecto	15. El Sistema señala los errores y evita el registro o modificación del usuario
	19. Clic en salir	
PostCondiciones	Se guardó la información exitosamente de un usuario nuevo o existente	
Requisito Funcional	RF-004	

- **Tabla 27 Guardar Trabajador**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Inhabilitar Trabajador</b>
Descripción		Permite inhabilitar a un usuario sin eliminar sus datos
Actor		Administrador
Precondiciones		El usuario debe estar logueado en el sistema
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El actor ingresa al menú de Gestión/trabajador	
	2. El actor el usuario deseado	
	3. El actor da clic en inhabilitar	4. El sistema pide confirmación
	5. El actor confirma la acción	6. El sistema cambia el estado del usuario a inactivo y muestra un mensaje de éxito.
	12. Clic en salir	
<b>Flujo alternativo</b>		

Pasos	Actor	Sistema
	13. El actor cancela la acción	14. El Sistema mantiene el estado del usuario sin realizar ningún cambio
	15. Clic en salir.	
PostCondiciones	Se cambio el estado del Usuario a inhabilitado).	
Requisito Funcional	RF-005	

- **Tabla 28 Inhabilitar Trabajador**

Fuente: Elaboración propia

Nombre		Crear Registro de tips	
Descripción		Permite registrar la información de los tips del banner con los siguientes datos:.	
Actor		Administrador.	
Precondiciones		El usuario debe estar logueado en el sistema	
Flujo básico			
Pasos	Actor	Sistema	
	1. El usuario ingresa al menú de Gestión / Registro de tips	2. El sistema despliega la interfaz con las opciones: Código, Título, Descripción, Prioridad.	
	3. El usuario da clic en botón Crear.	4. El sistema activa todos los campos y envía el cursor a código y activa los botones cancelar y guardar.	

	5. El usuario ingresa el código y da enter.	El sistema verifica si el código existe.
	6. El usuario ingresa los demás datos solicitados y va dando enter para llenarlos todos.	7. El sistema valida los campos requeridos
	8. El usuario da clic en guardar.	9. El sistema almacena y arroja un mensaje de transacción exitosa.
	10. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		10. El sistema confirma que hay inconsistencia en los datos y arroja el mensaje de alerta
	11. El usuario da clic en el botón cancelar	12. El sistema ubica el cursor en código
	13. El usuario ingresa los datos correctamente y da clic en guardar.	14. El sistema verifica los datos nuevamente.
		15. El sistema almacena y arroja un mensaje de transacción exitosa.
	16. Clic en salir	
PostCondiciones	Existe un nuevo tip registrado.	
Requisito Funcional	RF-015	

- **Tabla 29 Crear Registro de Tips**

Fuente: Elaboración propia

Nombre	Consultar Registro de tips
Descripción	Permite consultar la información de los tips registrados en la aplicación por el campo código.

Actor	Trabajador, SST y administrador.	
Precondiciones	El usuario debe estar logueado en el sistema	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión / Registro de tips	2. El sistema despliega la interfaz con las opciones: Código, Título, Descripción, Prioridad.
	3. El usuario da clic en botón Consultar.	4. El sistema activa el campo código para que el usuario realice la búsqueda. Activa los botones modificar, inhabilitar y cancelar.
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los datos en estado inactivo.
	8. Clic en salir	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		9. El sistema confirma que los datos ingresados no son correctos y arroja un mensaje de alerta.
	10. El usuario ingresa los datos correctamente y da enter	11. El sistema valida los datos nuevamente
		12. El sistema muestra los datos y todos los campos en estado inactivo.
	13. Clic en Salir	16. Clic en salir
PostCondiciones	Se realizó una consulta exitosa de los registros de tips existentes.	
Requisito Funcional	RF-016	

- **Tabla 30 Consultar Registro de tips**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Modificar Registro de tips</b>
Descripción		Permite realizar modificaciones a los atributos de los tips registrados en la aplicación a excepción del campo clave.
Actor		SST, Administrador
Precondiciones		El usuario debe estar logueado en el sistema y además debe haber realizado previamente la consulta del tip a modificar.
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión/ Registro de tips	2. El sistema despliega la interfaz con las opciones: Código, Título, Descripción, Prioridad, deshabilitados, con los botones crear, consultar, y salir activos, y los demás inactivos.
	3. El usuario da clic en botón consultar	4. El sistema activa el campo código para que el usuario realice la búsqueda y activa los botones modificar, inhabilitar, cancelar.
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los campos en estado inactivo.
	8. El usuario da clic en el botón modificar.	9. El Sistema activa los campos Código, Título, Descripción, Prioridad., envía el cursor al campo título y activa el botón guardar.
	10. El usuario modifica los datos y va dando enter	11. El sistema valida los datos

	12. El usuario da clic en el botón guardar	13. El sistema almacena y arroja un mensaje de transacción exitosa.
	14. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		15. El Sistema confirma que hay inconsistencias en los datos y arroja el mensaje de alerta.
	16. El usuario ingresa los datos correctamente y da clic en guardar.	17. El sistema confirma los datos nuevamente.
		18. El sistema almacena y arroja un mensaje de transacción exitosa.
	19. Clic en salir	
PostCondiciones	Se modifica un registro de tip existente.	
Requisito Funcional	RF-017	

- **Tabla 31 Modificar Registro de tips**

Fuente: Elaboración propia

<b>Nombre</b>	<b>Guardar Registro de tips</b>	
Descripción	Permite guardar los cambios realizados en los atributos de un tip previamente consultado.	
Actor	SST, Administrador	
Precondiciones	El usuario debe estar logueado en el sistema.	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión/ Registro de tips	2. El sistema despliega la interfaz con las opciones: Código, Título, Descripción, Prioridad, deshabilitados, con los botones crear, consultar, y salir activos, y los demás inactivos.

	3. El usuario da clic en botón consultar	4. El sistema activa el campo código para que el usuario realice la búsqueda y activa los botones modificar, inhabilitar, cancelar.
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los campos en estado inactivo.
	8. El usuario da clic en el botón modificar.	9. El Sistema activa los campos Código, Título, Descripción, Prioridad, envía el cursor al campo nombres y activa el botón guardar.
	10. El usuario modifica los datos y va dando enter	11. El sistema valida los datos
	12. El usuario da clic en el botón guardar	13. El sistema almacena y arroja un mensaje de transacción exitosa.
	14. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		15. El Sistema confirma que hay inconsistencias en los datos y arroja el mensaje de alerta.
	16. El usuario ingresa los datos correctamente y da clic en guardar.	17. El sistema confirma los datos nuevamente.
		18. El sistema almacena y arroja un mensaje de transacción exitosa.
	19. Clic en salir	
PostCondiciones	Se guardó la información exitosamente relacionada con el tip	

Requisito Funcional	RF-018	
---------------------	--------	--

- **Tabla 32 Guardar Registro de tips**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Inhabilitar Registro de tips</b>	
Descripción		Permite inhabilitar o habilitar un tip previamente registrado.	
Actor		Administrador	
Precondiciones		El usuario debe estar logueado en el sistema	
<b>Flujo básico</b>			
Pasos	Actor	Sistema	
	1. El usuario ingresa al menú de Gestión/ Registro de tips	2. El sistema despliega la interfaz con las opciones: Código, Título, Descripción, Prioridad, deshabilitados, con los botones crear, consultar, y salir activos, y los demás inactivos.	
	3. El usuario da clic en botón consultar	4. El sistema activa el campo código para que el usuario realice la búsqueda y activa los botones modificar, inhabilitar, cancelar.	
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.	
		7. El sistema muestra los datos y todos los campos en estado inactivo.	
	8. El usuario da clic en el botón inhabilitar.	9. El sistema muestra advertencia y valida si es seguro de inhabilitar ese tip, dando opción de sí/no	
	10. El usuario selecciona la opción sí	11. El sistema cambia el estado del tip y arroja un mensaje de transacción exitosa.	
	12. Clic en salir		
	<b>Flujo alternativo</b>		
		Actor	Sistema

Pasos	13. El usuario selecciona la opción no	14. El Sistema muestra los datos y todos los campos en estado activo. El sistema inactiva los botones modificar e inhabilitar.
	15. Clic en salir.	
PostCondiciones	Se cambio el estado del tip habilitado a inhabilitado).	
Requisito Funcional	RF-019	

- **Tabla 33 Inhabilitar Registro de tips**

Fuente: Elaboración propia

<b>Crear Temperatura corporal</b>		
<b>Nombre</b>		
Descripción	Permite registrar la temperatura corporal actual de un trabajador con los siguientes datos: cédula, fecha y hora.	
Actor	Administrador.	
Precondiciones	El usuario debe estar logueado en el sistema	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión / Temperatura corporal	2. El sistema despliega la interfaz con las opciones: Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal.
	3. El usuario da clic en botón Crear.	4. El sistema activa todos los campos y envía el cursor a código y activa los botones cancelar y guardar.
	5. El usuario ingresa el código y da enter.	El sistema verifica si el código existe.
	6. El usuario ingresa los demás datos	7. El sistema valida los campos requeridos

	solicitados y va dando enter para llenarlos todos.	
	8. El usuario da clic en guardar.	9. El sistema almacena y arroja un mensaje de transacción exitosa.
	10. Clic en salir.	
<b>Flujo alternativo</b>		
	Actor	Sistema
		10. El sistema confirma que hay inconsistencia en los datos y arroja el mensaje de alerta
Pasos	11. El usuario da clic en el botón cancelar	12. El sistema ubica el cursor en código
	13. El usuario ingresa los datos correctamente y da clic en guardar.	14. El sistema verifica los datos nuevamente.
		15. El sistema almacena y arroja un mensaje de transacción exitosa.
	16. Clic en salir	
PostCondiciones	Existe una nueva Temperatura corporal registrada en el sistema.	
Requisito Funcional	RF-015	

- **Tabla 34 Crear Temperatura corporal**

Fuente: Elaboración propia

<b>Nombre</b>	<b>Consultar Temperatura corporal</b>
---------------	---------------------------------------

Descripción	Permite consultar la información de la Temperatura corporal registrada en la aplicación por el campo código.	
Actor	Trabajador, SST y administrador.	
Precondiciones	El usuario debe estar logueado en el sistema	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión / Temperatura corporal	2. El sistema despliega la interfaz con las opciones: Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal.
	3. El usuario da clic en botón Consultar.	4. El sistema activa el campo código para que él usuario realice la búsqueda. Activa los botones modificar, inhabilitar y cancelar.
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los datos en estado inactivo.
	8. Clic en salir	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		9. El sistema confirma que los datos ingresados no son correctos y arroja un mensaje de alerta.
	10. El usuario ingresa los datos correctamente y da enter	11. El sistema valida los datos nuevamente
	12. El sistema muestra los datos y todos los campos en estado inactivo.	

PostCondiciones	13. Clic en Salir	16. Clic en salir
	Se realizó una consulta exitosa de los registros de temperatura corporal existentes.	
Requisito Funcional	RF-016	

- **Tabla 35 Consultar Temperatura corporal**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Modificar Temperatura corporal</b>
Descripción		Permite realizar modificaciones a los atributos de un registro de temperatura corporal previamente consultado en la aplicación a excepción del campo clave.
Actor		Administrador
Precondiciones		El usuario debe estar logueado en el sistema y además debe haber realizado previamente el registro de temperatura corporal que desea modificar.
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión/ Temperatura corporal	2. El sistema despliega la interfaz con las opciones: Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal, deshabilitados, con los botones crear, consultar, y salir activos, y los demás inactivos.
	3. El usuario da clic en botón consultar	4. El sistema activa el campo código para que el usuario realice la búsqueda y activa los botones modificar, inhabilitar, cancelar.

	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los campos en estado inactivo.
	8. El usuario da clic en el botón modificar.	9. El Sistema activa los campos Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal, envía el cursor al campo código y activa el botón guardar.
	10. El usuario modifica los datos y va dando enter	11. El sistema valida los datos
	12. El usuario da clic en el botón guardar	13. El sistema almacena y arroja un mensaje de transacción exitosa.
	14. Clic en salir.	
<b>Flujo alternativo</b>		
	Actor	Sistema
Pasos		15. El Sistema confirma que hay inconsistencias en los datos y arroja el mensaje de alerta.
	16. El usuario ingresa los datos correctamente y da clic en guardar.	17. El sistema confirma los datos nuevamente.
		18. El sistema almacena y arroja un mensaje de transacción exitosa.
PostCondiciones	19. Clic en salir	
	Se modifica un registro existente de temperatura	

	corporal correctamente.	
Requisito Funcional	RF-017	

- **Tabla 36 Modificar Temperatura corporal**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Guardar Temperatura corporal</b>
Descripción		Permite guardar los cambios realizados en los atributos de una temperatura corporal previamente consultada.
Actor		Administrador
Precondiciones		El usuario debe estar logueado en el sistema.
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión/ Temperatura corporal	2. El sistema despliega la interfaz con las opciones: Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal, deshabilitados, con los botones crear, consultar, y salir activos, y los demás inactivos.
	3. El usuario da clic en botón consultar	4. El sistema activa el campo código para que el usuario realice la búsqueda y activa los botones modificar, inhabilitar, cancelar.
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los campos en estado inactivo.

	8. El usuario da clic en el botón modificar.	9. El Sistema activa los campos Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal, envía el cursor al campo nombres y activa el botón guardar.
	10. El usuario modifica los datos y va dando enter	11. El sistema valida los datos
	12. El usuario da clic en el botón guardar	13. El sistema almacena y arroja un mensaje de transacción exitosa.
	14. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		15. El Sistema confirma que hay inconsistencias en los datos y arroja el mensaje de alerta.
	16. El usuario ingresa los datos correctamente y da clic en guardar.	17. El sistema confirma los datos nuevamente.
		18. El sistema almacena y arroja un mensaje de transacción exitosa.
PostCondiciones	19. Clic en salir	
	Se guardó la información exitosamente relacionada con el registro de temperatura corporal.	
Requisito Funcional	RF-018	

- **Tabla 37 Guardar Temperatura corporal**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Inhabilitar Temperatura corporal</b>
Descripción	Permite inhabilitar o habilitar un registro de temperatura corporal previamente registrado.	
Actor	Administrador	
Precondiciones	El usuario debe estar logueado en el sistema	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Gestión/ Temperatura corporal	2. El sistema despliega la interfaz con las opciones: Código, Cédula, Nombres, Apellidos, Fecha, Hora, Temperatura corporal, deshabilitados, con los botones crear, consultar, y salir activos, y los demás inactivos.
	3. El usuario da clic en botón consultar	4. El sistema activa el campo código para que el usuario realice la búsqueda y activa los botones modificar, inhabilitar, cancelar.
	5. El usuario ingresa el código y da enter	6. El sistema verifica si el código existe.
		7. El sistema muestra los datos y todos los campos en estado inactivo.
	8. El usuario da clic en el botón inhabilitar.	9. El sistema muestra advertencia y valida si es seguro de inhabilitar ese registro, dando opción de sí/no
	10. El usuario selecciona la opción sí	11. El sistema cambia el estado del registro y arroja un mensaje de transacción exitosa.
	12. Clic en salir	
<b>Flujo alternativo</b>		
	Actor	Sistema

Pasos	13. El usuario selecciona la opción no	14. El Sistema muestra los datos y todos los campos en estado activo. El sistema inactiva los botones modificar e inhabilitar.
PostCondiciones	15. Clic en salir. Se cambio el estado del registro de temperatura corporal habilitado a inhabilitado).	
Requisito Funcional	RF-019	

- **Tabla 38 Inhabilitar Temperatura corporal**

Fuente: Elaboración propia

Nombre		Informe de usuarios registrados
Descripción		Permite a los actores generar un informe con el historial de usuarios registrados en el sistema, incluyendo datos personales, estado activo/inactivo y perfil asignado.
Actor		SST, administrador.
Precondiciones		El usuario debe estar logueado en el sistema.
Flujo básico		
Pasos	Actor	Sistema
	1. El usuario accede al menú Salidas / Informes / Generar informes de usuarios registrados.	2. El sistema despliega la interfaz con las opciones: Cédula, nombre y estado.

	3. El usuario da clic en botón Generar informe	4. El sistema activa todos los campos requeridos, el cursor apunta en "cédula".
	5. El usuario ingresa los filtros deseados.	6. El sistema valida información ingresada
	7. El usuario da clic en guardar.	8. El sistema genera el informe y muestra un mensaje de transacción exitosa.
	9. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		10. El sistema detecta inconsistencia en los datos ingresados.
	11. muestra un mensaje de alerta.	12. El usuario da clic en "Cancelar".
	13. El sistema limpia el formulario y el cursor queda en el campo "Cedula"	14. El usuario corrige los datos y da clic en "Guardar".
	15. El sistema valida y genera el informe nuevamente.	16. El sistema muestra un mensaje de transacción exitosa.
PostCondiciones	El informe de usuarios registrados queda almacenado y disponible para consulta o descarga.	
Requisito Funcional	RF-072	

- **Tabla 39 Informe de usuarios registrados**

Fuente: Elaboración propia

<b>Nombre</b>		<b>Informe de activación de alertas críticas</b>
Descripción	Permite a los actores generar un informe con el historial de alertas críticas generadas cuando los sensores del casco detectan condiciones térmicas altas.	
Actor	SST, administrador.	
Precondiciones	El usuario debe estar logueado en el sistema.	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario accede al menú salidas / Informes / Alertas críticas.	2. El sistema muestra los campos: rango de fechas.
	3. El usuario da clic en "Generar informe".	4. El sistema activa los campos e inicia en "Fecha".
	5. El usuario ingresa los rangos de fechas.	6. El sistema valida los datos y consulta el historial de alertas.
	7. El usuario da clic en guardar.	8. El sistema genera y almacena el informe. Muestra mensaje de éxito.
	9. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		10. El sistema muestra un mensaje de error por inconsistencias.
	11. El usuario da click en el botón cancelar	12. El sistema posiciona el cursor en "Fecha".
	13. El usuario corrige los datos	14. El sistema verifica y confirma los datos.

	y da clic en "Guardar".	
	15. El informe se almacena exitosamente.	16. Clic en salir
PostCondiciones	El informe de alertas críticas queda generado y disponible para consulta o descarga.	
Requisito Funcional	RF-074	

- **Tabla 40 Informe de activación de alertas críticas**

<b>Nombre</b>		<b>Informe estado de la temperatura de los usuarios</b>
Descripción	Permite generar un informe con el historial de pagos imputados por los servicios prestados.	
Actor	SST y Administrador.	
Precondiciones	El usuario debe estar logueado en el sistema.	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario ingresa al menú de Salidas / Informes / temperatura de los usuarios	2. El sistema despliega la interfaz con las opciones: Rangos de fecha.
	3. El usuario da clic en botón crear	4. El sistema activa la opción de rangos de fecha y botones cancelar y guardar.
	5. El usuario ingresa el rango de fechas y da enter.	6. El sistema valida los campos requeridos.

	7. El actor da clic en guardar.	8. El sistema almacena y arroja un mensaje de transacción exitosa.
	9. Clic en salir.	
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		10. El sistema detecta inconsistencias y muestra mensaje de alerta.
	11. El usuario da clic en "Cancelar".	12. El sistema ubica el cursor nuevamente en el campo 'Fecha'.
	13 el usuario corrige los datos y da clic en "Guardar".	14. El sistema verifica y confirma los datos.
	15. El sistema almacena el informe exitosamente.	16. Clic en salir
PostCondiciones	El informe de estado de temperatura queda registrado y disponible.	
Requisito Funcional	RF-073	

- **Tabla 41 Informe estado de la temperatura de los usuarios**

Fuente: Elaboración propia

Nombre	Consultar ubicación en panel de monitoreo
Descripción	Permite visualizar en tiempo real la ubicación del trabajador mediante el

	módulo GPS del casco, desde un panel de monitoreo.	
Actor	SST, Administrador	
Precondiciones	El usuario debe estar logueado en el sistema	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario accede al menú salidas / Consultas / consultar la ubicación en panel de monitoreo.	2. El sistema despliega el panel con el campo de búsqueda por cédula o nombre del trabajador.
	3. El usuario ingresa la cédula o nombre del trabajador.	4. El sistema localiza al trabajador y muestra su ubicación en el mapa.
	5. El usuario puede visualizar detalles como hora, temperatura y nivel de alerta si aplica.	6. El sistema permite tomar captura o generar informe si es necesario.
	7. Clic en salir.	

Flujo alternativo		
Pasos	Actor	Sistema
		8. El sistema muestra mensaje de error si la cédula no está registrada o el casco no transmite datos.
	9. El usuario ingresa nuevamente la información o da clic en "Cancelar".	11. El sistema reinicia el campo de búsqueda.
PostCondiciones	Se muestra la ubicación del trabajador en el mapa con la información relacionada.	
Requisito Funcional	RF-064	

- **Tabla 42 Consultar ubicación en panel de monitoreo**

Fuente: Elaboración propia

Nombre	Consultar tips en el banner de la app
Descripción	Permite a los actores visualizar tips de seguridad relacionados con la prevención de golpes de calor, mostrados en el banner de la aplicación móvil.

Actor	Trabajador, SST, Administrador	
Precondiciones	El usuario debe estar logueado en la aplicación	
<b>Flujo básico</b>		
Pasos	Actor	Sistema
	1. El usuario accede a la aplicación desde su dispositivo móvil.	2. El sistema carga automáticamente el banner de tips en la pantalla principal.
	3. El usuario puede deslizar para ver más tips o tocar para ampliar.	4. El sistema muestra la información completa del tip seleccionado.
	5. El sistema permite regresar a la pantalla principal o cerrar el banner.	6. Clic en salir.
<b>Flujo alternativo</b>		
Pasos	Actor	Sistema
		7. El sistema detecta que no hay tips registrados o activos.
	8. Se muestra un mensaje: "No hay recomendaciones"	

	disponibles en este momento.”	
PostCondiciones	El usuario accede a los tips de seguridad actualizados en el banner de la app.	
Requisito Funcional	RF-065	

- **Tabla 43 Consultar tips en el banner de la app**

Fuente: Elaboración propia

Nombre	Consultar historial de alertas	
Descripción	Permite consultar el historial de alertas generadas por el casco inteligente cuando se detectan condiciones térmicas críticas, mostrando los registros por trabajador y fecha	
Actor	SST, Administrador	
Precondiciones	El usuario debe estar logueado en el sistema	
Flujo básico		
	Actor	Sistema

Pasos	1. El usuario accede al menú Salidas / Consultas / Historial de alertas.	2. El sistema despliega la interfaz con los campos: cédula del trabajador y rango de fechas.
	3. El usuario ingresa la cédula y selecciona el rango de fechas a consultar.	4. El sistema valida los datos ingresados.
	5. El usuario da clic en "Consultar".	6. El sistema muestra el listado de alertas correspondientes al trabajador, incluyendo fecha, hora, nivel de alerta, temperatura y ubicación.
	7. El usuario puede visualizar el detalle de cada alerta.	8. El sistema permite generar un informe o exportar los resultados.
	9. Clic en "Salir".	
<b>Flujo alternativo</b>		
	Actor	Sistema
		10. El sistema detecta que no hay registros en el rango indicado o que la cédula no está registrada.

Pasos	11.Muestra mensaje de alerta	12. El usuario da clic en "Cancelar" o modifica los datos.
PostCondiciones	Se presenta el historial de alertas consultado, con opción de generar informe o exportar.	
Requisito Funcional	RF-065	

- **Tabla 44 Consultar historial de alertas**

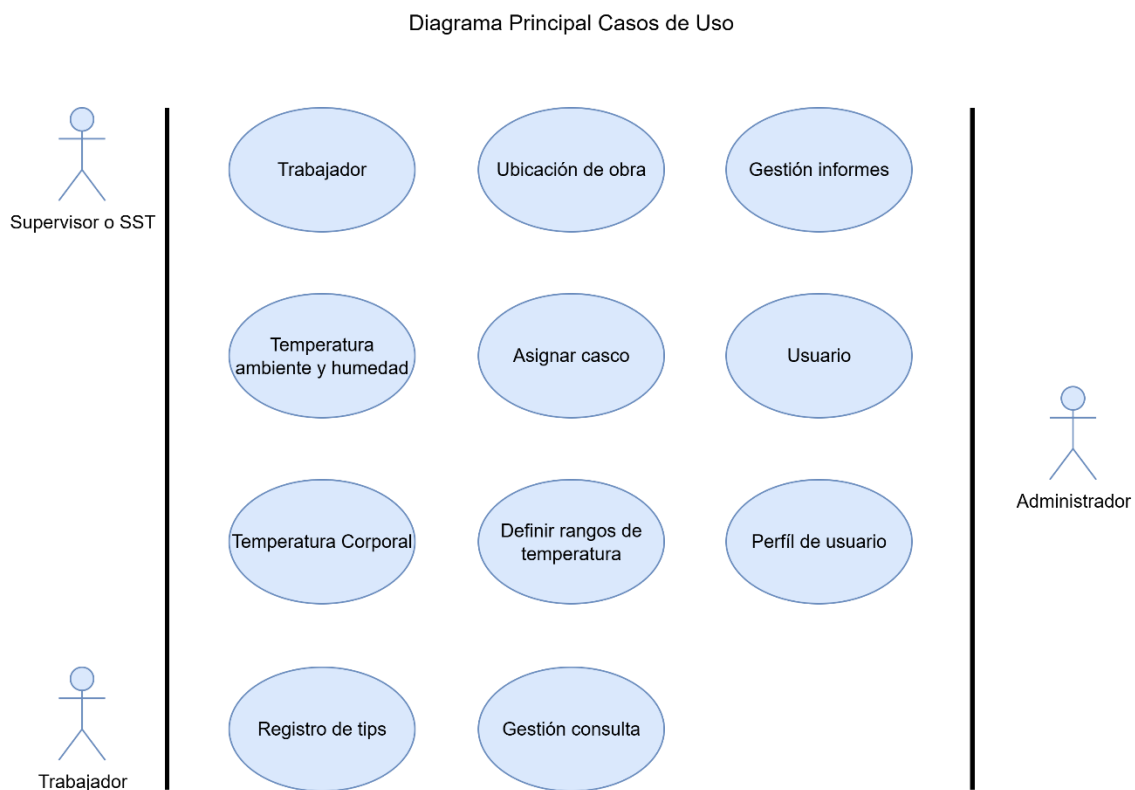
Fuente: Elaboración propia

## 16. Especificación del Requisito

“Los requerimientos/requisitos de un sistema describen los servicios que ha de ofrecer el sistema y las restricciones asociadas a su funcionamiento.

Propiedades o restricciones determinadas de forma precisa que deben satisfacerse.”  
(Granada, 2025)

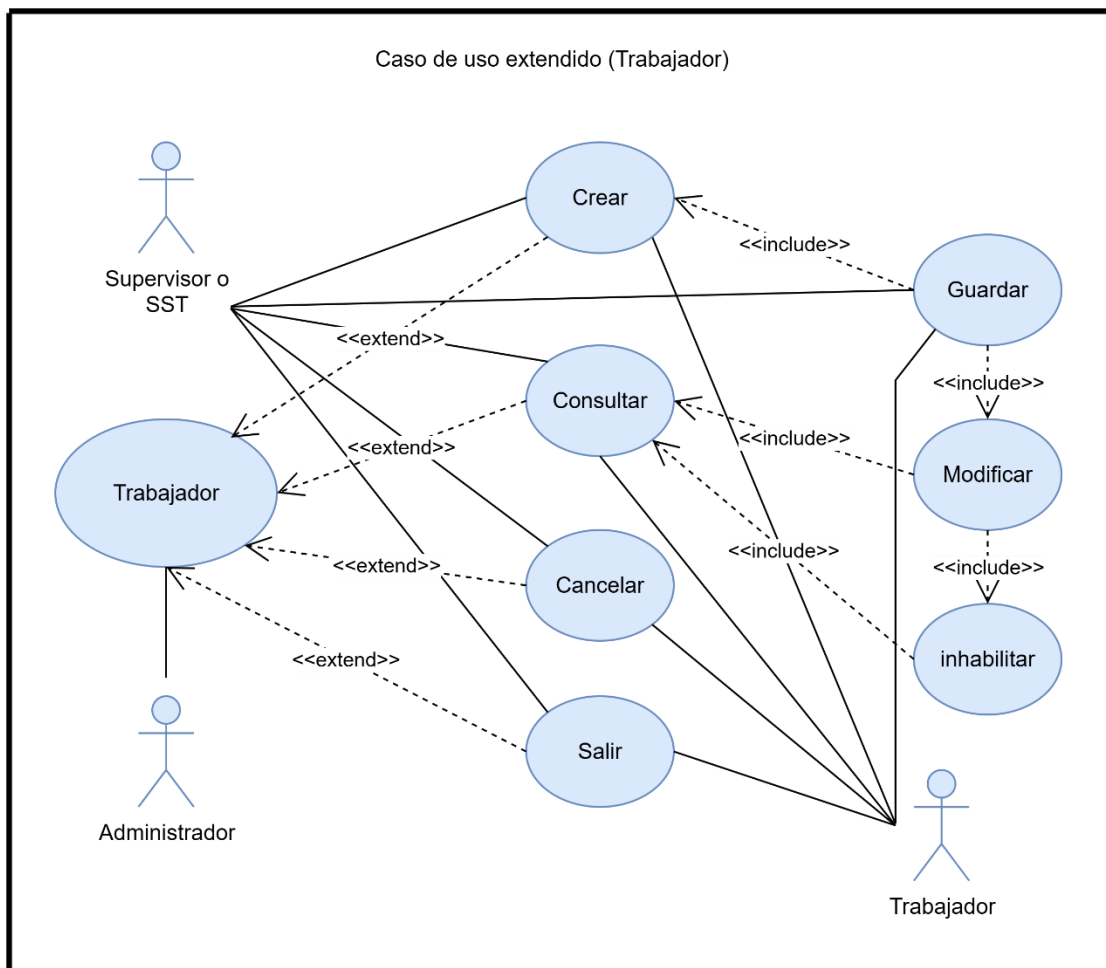
## 16.1 Diagrama de caso de uso principal



- **Figura 5 Diagrama principal casos de uso**

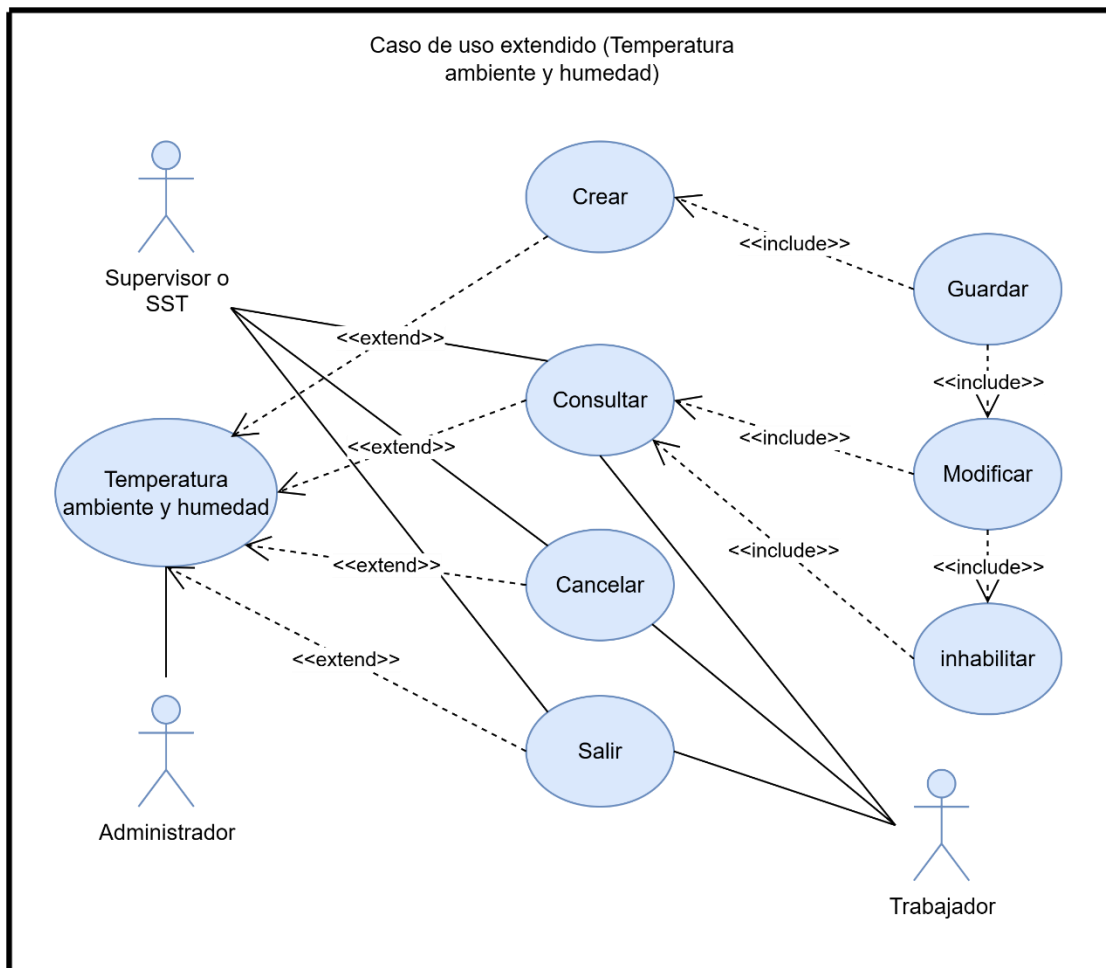
Fuente: elaboración propia.

## 16.2 Diagramas casos de uso extendidos



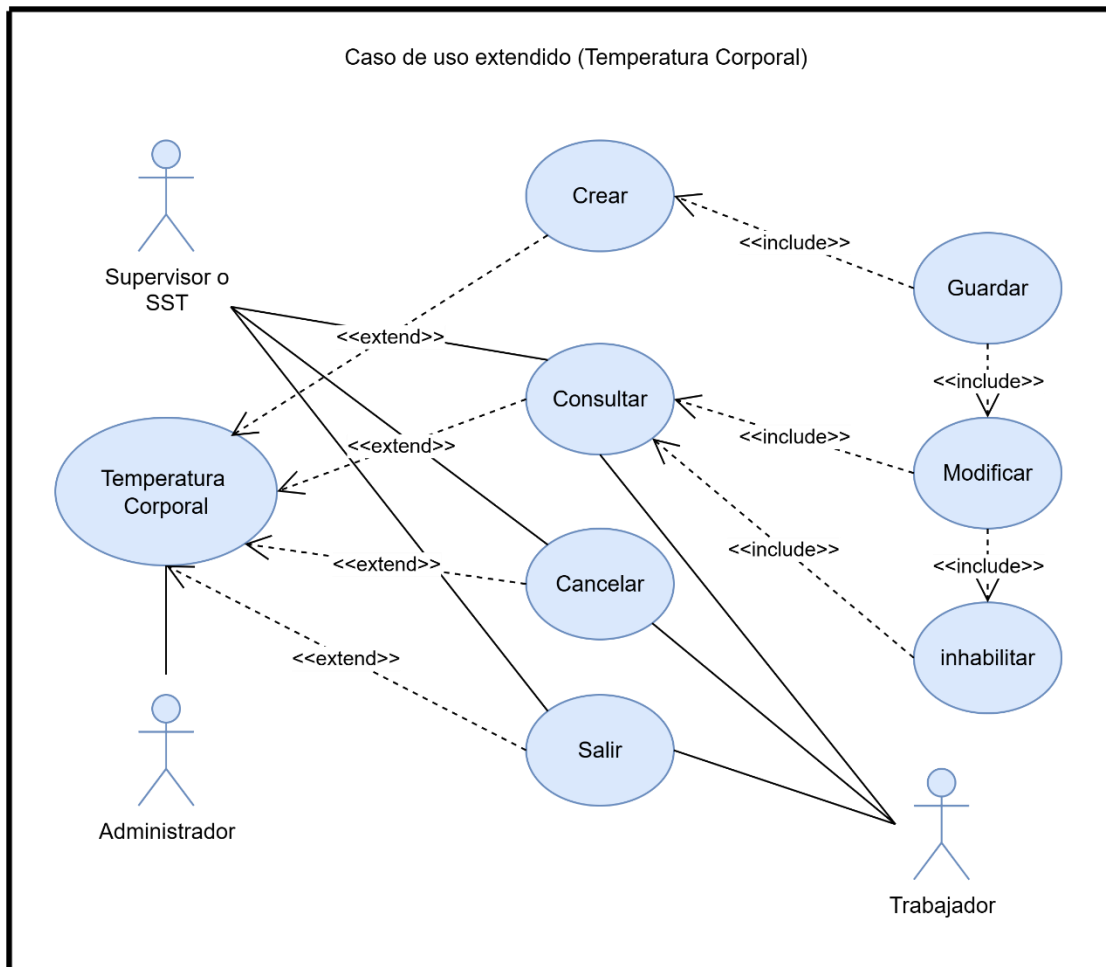
- **Figura 6 Casos de Usos Extendidos (trabajador)**

Fuente: elaboración propia.



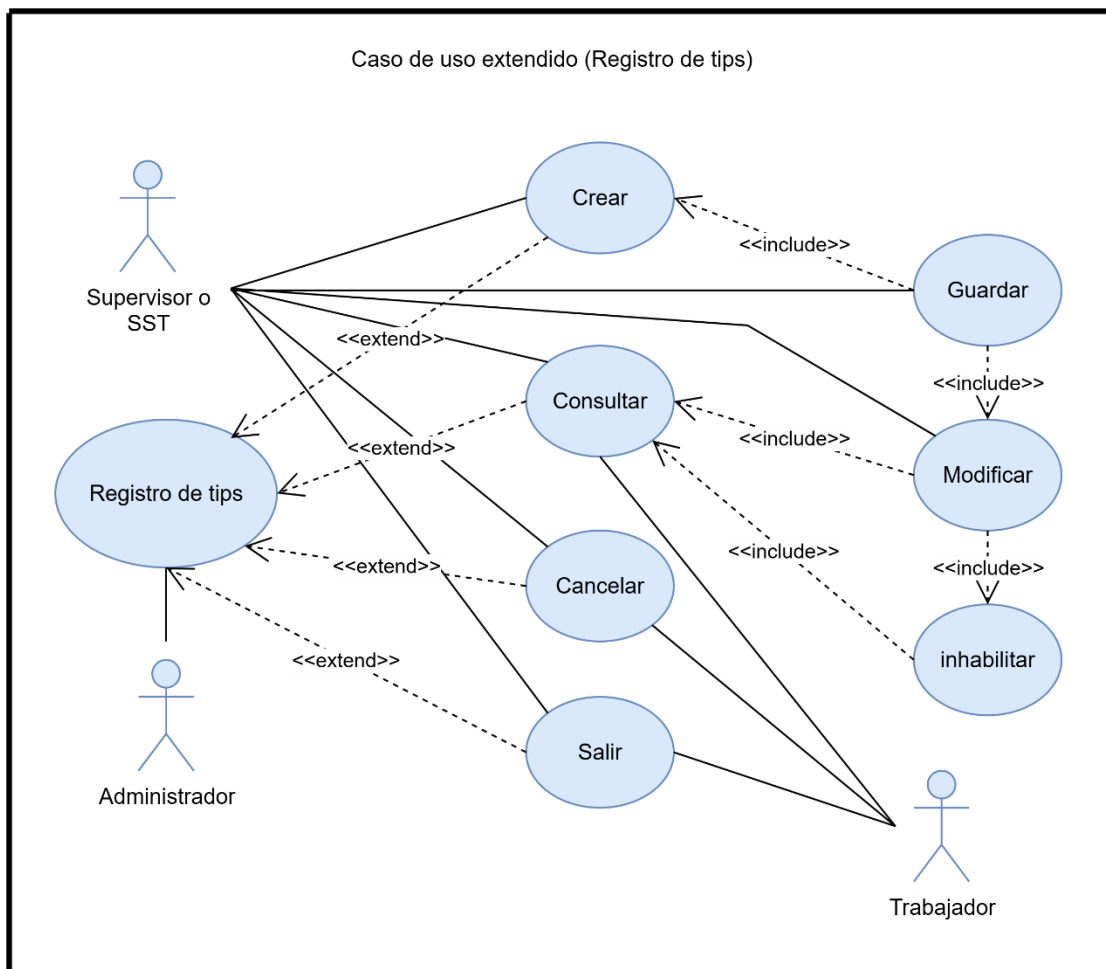
- **Figura 7 Casos de Uso Extendido (temperatura ambiente y humedad)**

Fuente: elaboración propia.



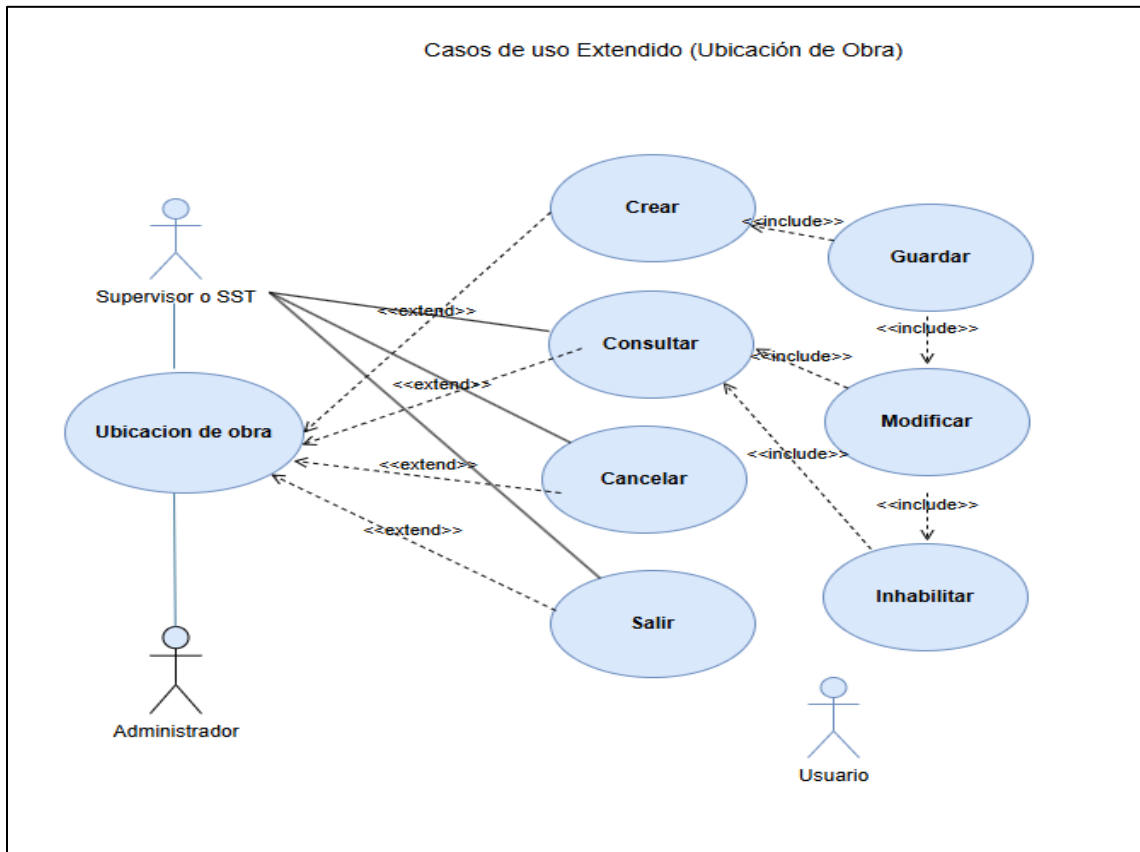
- **Figura 8 Casos de Uso Extendido (temperatura corporal)**

Fuente: elaboración propia.



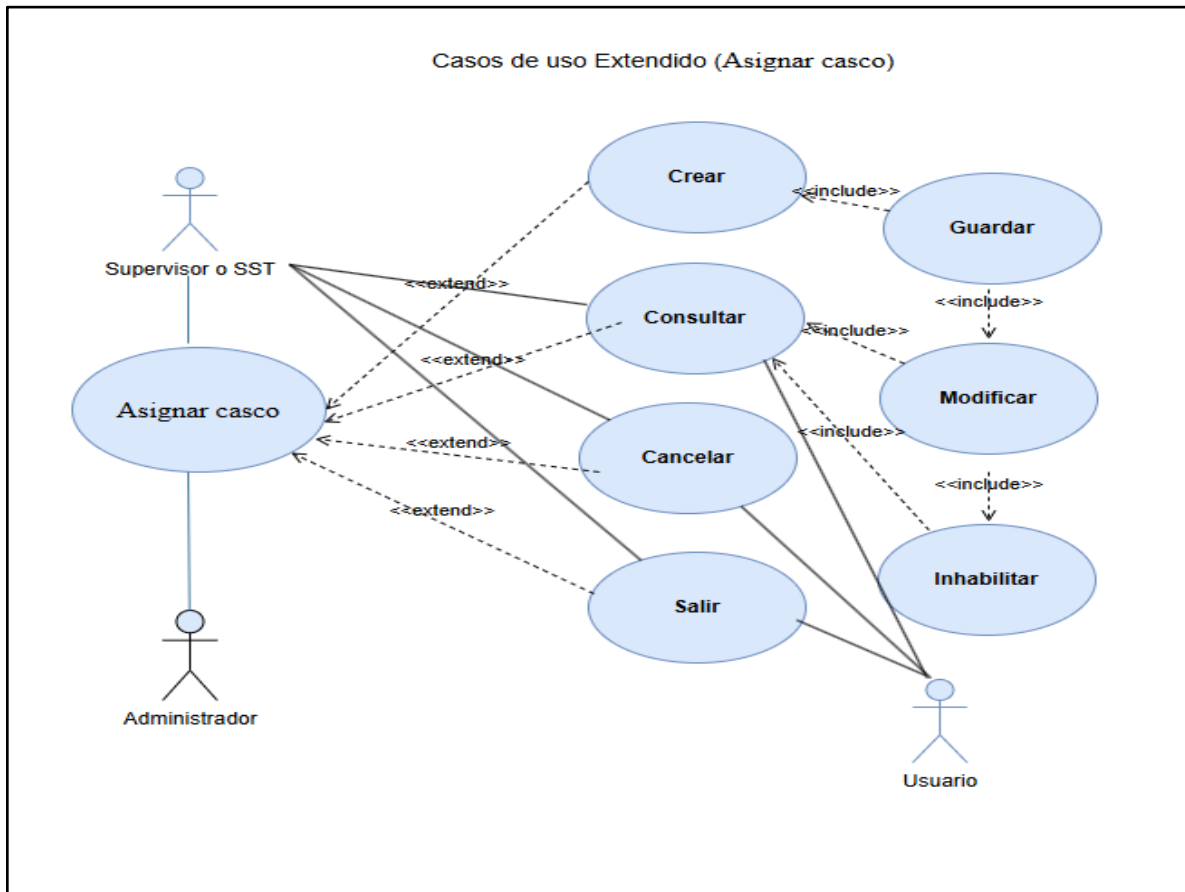
• **Figura 9 Casos de Uso Extendido (Registro de Tips)**

Fuente: elaboración propia.



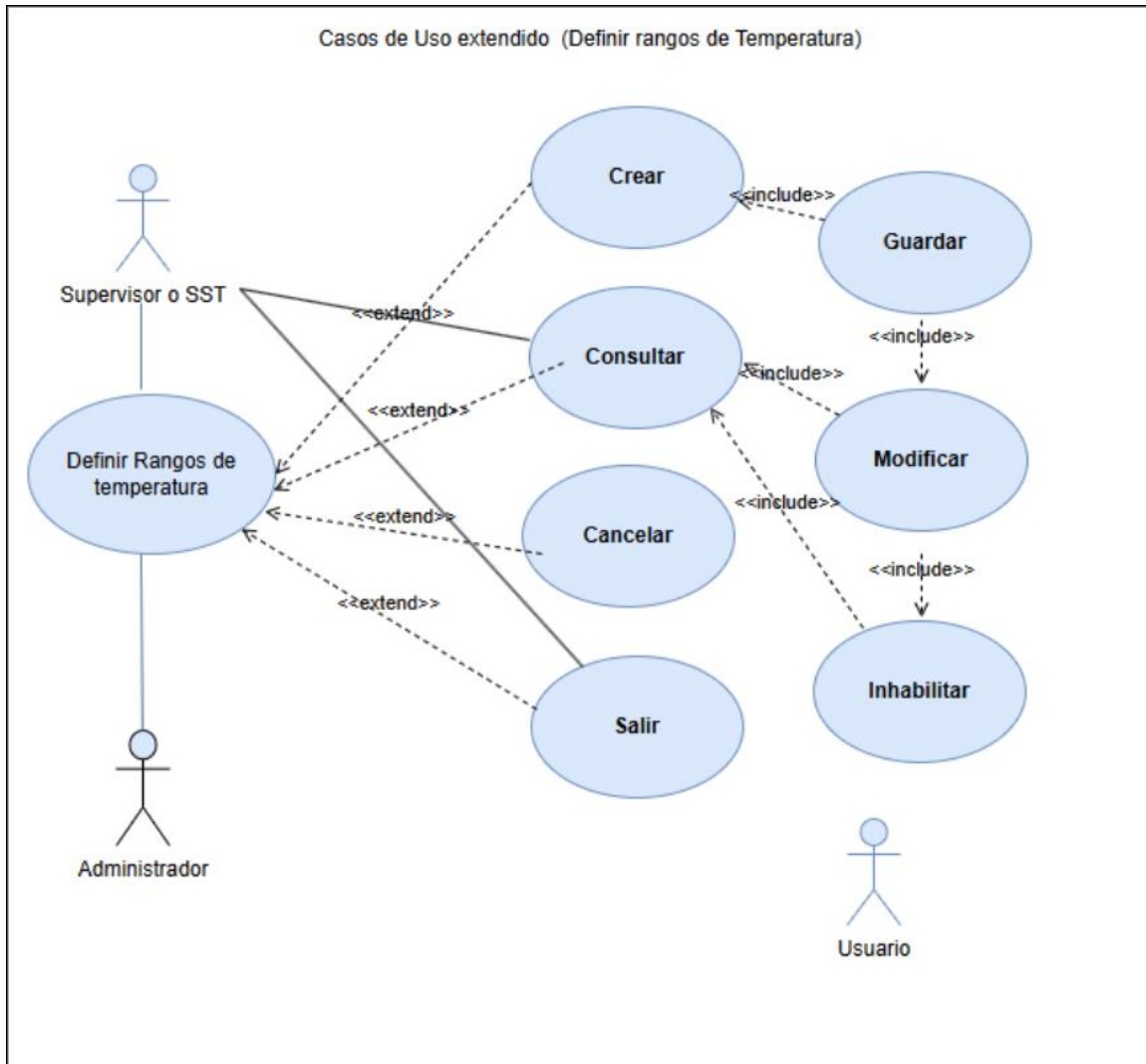
- **Figura 10 Casos de Uso Extendido (Ubicación de Obra)**

Fuente: Elaboración propia



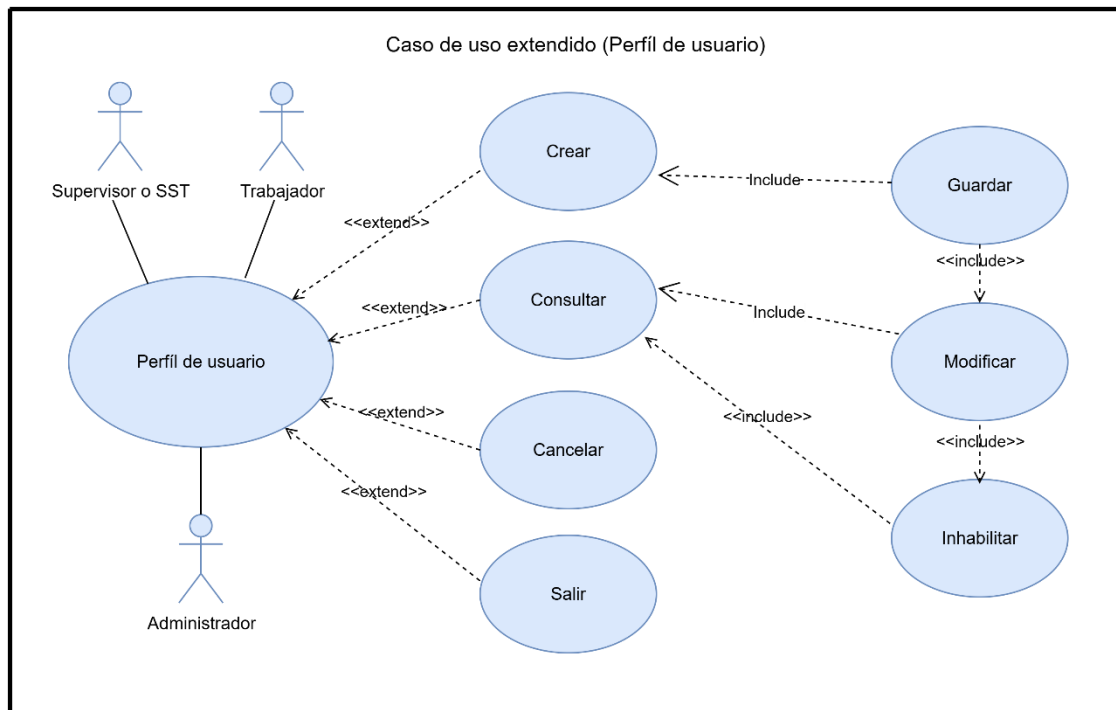
- **Figura 11 Casos de Uso Extendido (Asignar ca)**

Fuente: Elaboración propia



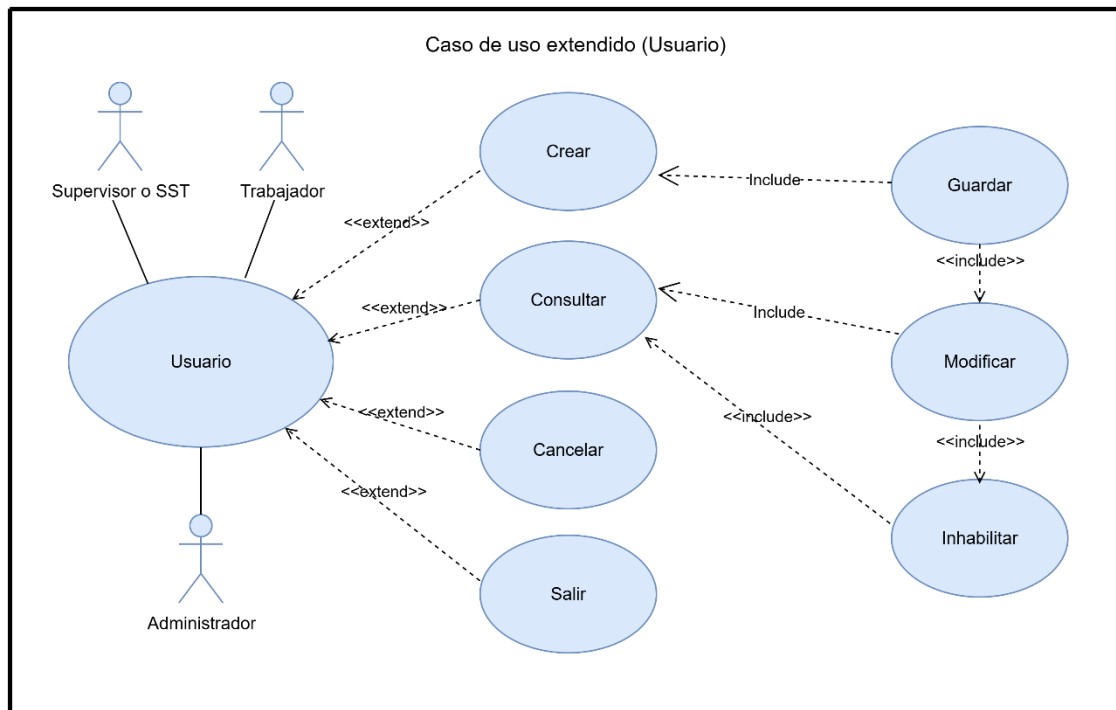
- **Figura 12 Casos de Uso Extendido (Definir rangos de Temperatura)**

Fuente: elaboración propia.



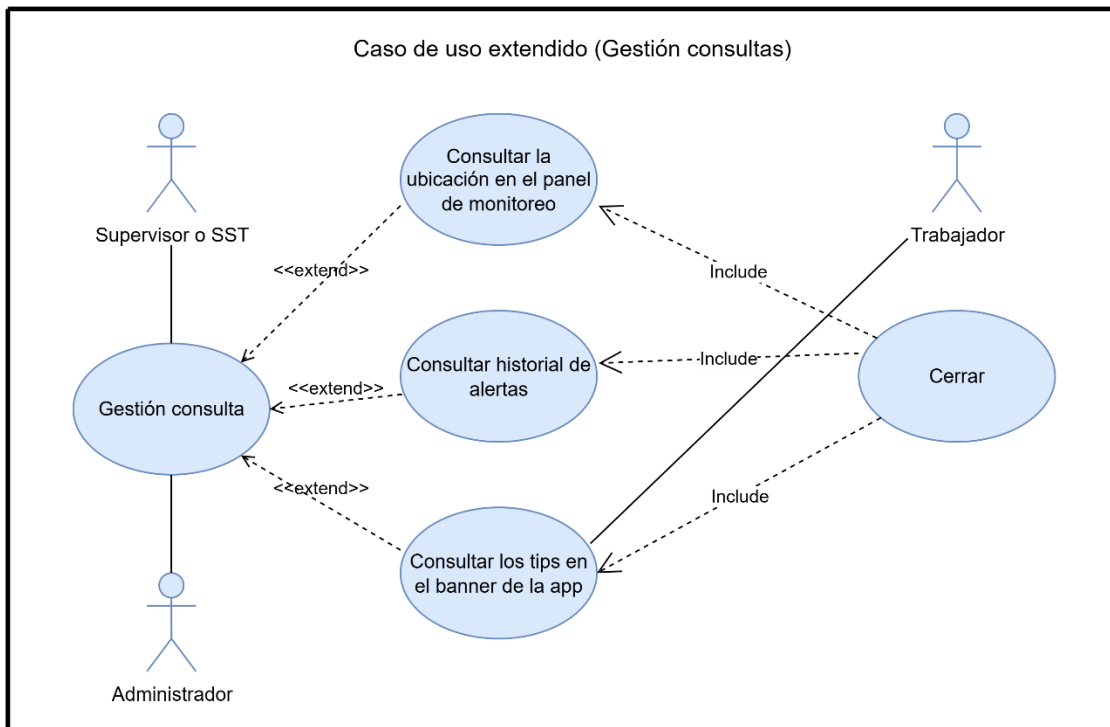
- **Figura 13 Casos de Uso Extendido (Perfil de Usuario)**

Fuente: elaboración propia.



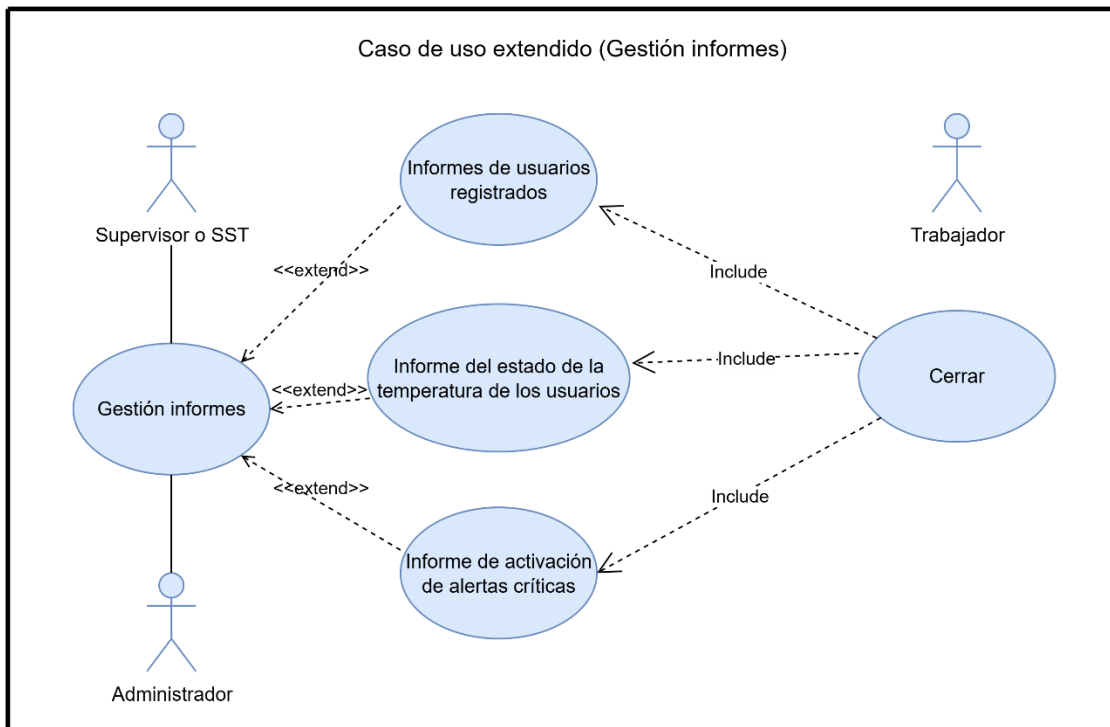
- **Figura 14 Casos de Uso Extendido (Usuario)**

Fuente: elaboración propia.



- **Figura 15 Casos de Uso Extendido (Gestión consulta)**

Fuente: elaboración propia.



- **Figura 16 Casos de Uso Extendido (Gestión Informe)**

Fuente: elaboración propia.

## 17. Mapa de Navegación

Un mapa de navegación es una representación visual de las páginas que contiene un sitio web y su información. De esta forma, se realiza un recorrido comprensible y facilita el acceso de cada contenido a los usuarios. Es importante mencionar que un sitio web debe brindar agilidad en su navegación, usabilidad para que los usuarios permanezcan más tiempo en él y brindando siempre una buena experiencia.

Este recurso es fundamental para la administración, ya que cada generador de contenido sabrá dónde debe quedar cada pieza de información y podrán realizarse procesos de revisión como las auditorías web con mayor facilidad.

También es útil como guía para crear el mapa XML, un archivo que es posible generar de manera automática (con herramientas como Yoast) y que puede enviarse en la Google Search Console para que tu sitio web esté indexado y aparezca en el buscador.

Características de un mapa de navegación web.

Contempla diferentes categorías de navegación

Navegación por estructura: relación entre el tema principal y los subtemas asociados a este. Por ejemplo: tema «Sobre nosotros» con subtemas «Trabaja con nosotros»

Asociativa: relaciona diversas páginas que tienen contenidos similares. Ejemplo: página de blog que conecta con una landing page de productos.

Utilitaria: integra contenidos que muchas veces son independientes de la estructura jerárquica. Ejemplo: «Políticas de privacidad» y «Contacto».

### 17.1 Tipos de mapas de navegación web.

Estructura lineal

Cada página está vinculada a la anterior, por lo que funciona bien en un sitio de pocas páginas.

### 17.2 Estructura jerárquica

Esta estructura pone en un rango superior la página de inicio, y en un nivel debajo de ella las diferentes categorías y subcategorías del sitio web.

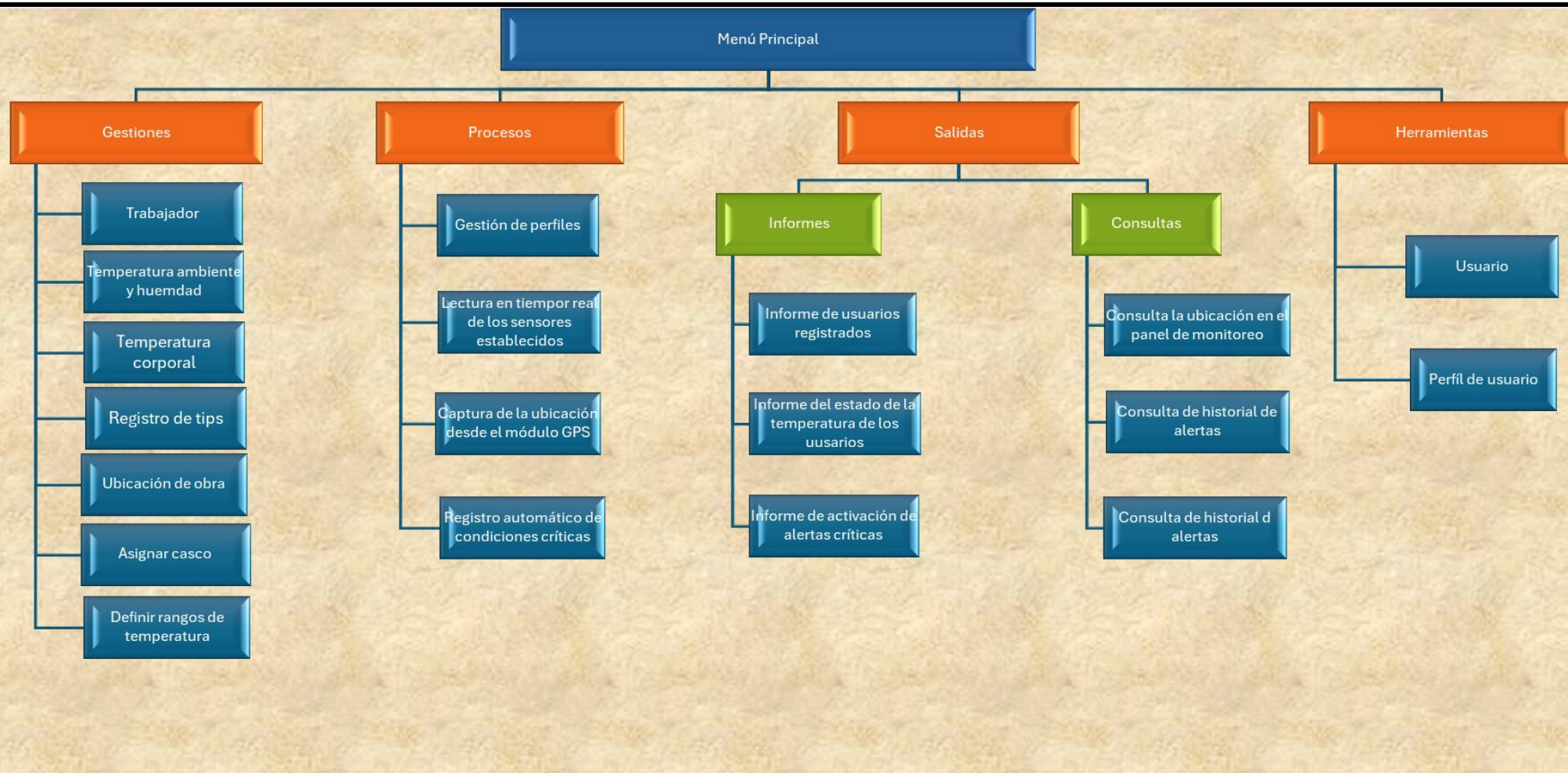
### 17.3 Estructura web

La navegación aquí les permite a los usuarios ir de una página a otra, sin necesariamente seguir un orden jerárquico y siguiendo su propio flujo de información.

#### **17.4 Estructura compuesta**

Esta se compone de varios tipos de estructura como la lineal, en donde alguna parte deriva a una estructura jerárquica, lo que permite jugar ampliamente con la organización de diversa información.” (Diseño y desarrollo web, n.d.)

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**  
**17.5. Mapa de Navegación**



• **Figura 17 Mapa Navegación**

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**

**19. Recursos (Hardware, Software, Talento Humano)**


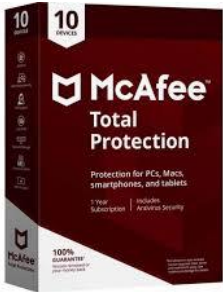
**19.1 Recurso Hardware**



Cantidad	Recurso 1	Imagen	Costo Unitario
3	Portátil Asus Vivobook Go E1504fa Amd Ryzen 5 7520u 16gb 1tb para desarrollo y pruebas		\$1.915.990
	Características	Con capacidad mínima de 8 GB de RAM, procesador i7	
3	Mouse inalámbrico recargable Yindiao A2 negro		\$42.079
3	Base Refrigerante Portátil 638b Plegable Negro Cooling Pad Con Un Ventilador		\$29.990

3	Teclados logitech pebble		\$ 179.900
2	Teléfonos Samsung Galaxy A05 para pruebas de conectividad y monitoreo		\$383.900

Fuente: <https://listado.mercadolibre.com.co/computador-portatil>

## 19.2 Recurso software

Recurso	Imagen	Costo Unitario
Licencia Office 365 1 Usuario 12 Meses		\$180.000 cop
McAfee Total Protection Antivirus 10 Dispositivos 1 Año		\$99.900 cop

Base de datos Firebase		150,73 US\$ Mensuales
GitHub		\$2.000 USD por mes

Fuente: <https://acortar.link/ckAtzN>

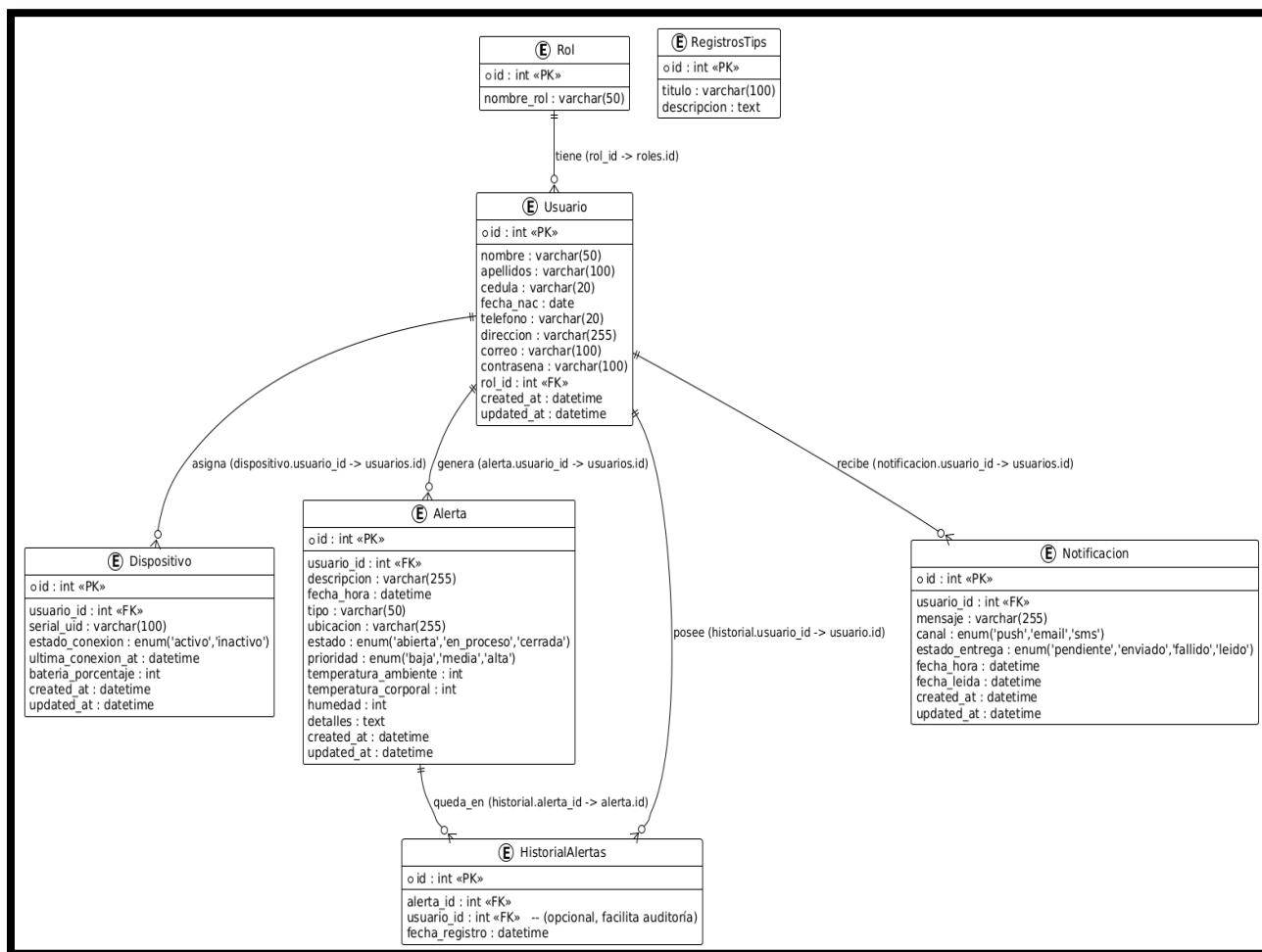
### 19.3 Recurso talento Humano

Cargo	Tiempo	Salario	Valor/Semana
Desarrollador UX y Tester (QA)	Permanente	3,500,000	875,000
Desarrollador Fronted y BD	Permanente	4,000,000	1,000,000
Desarrollador Backend (lider)	Permanente	6,000,000	1,500,000
Experto SG-SST	Asesoría	2,800,000	700,000
ING Electrónico	Temporal	5,000,000	1,250,000

Fuente: <https://co.computrabajo.com/salarios/programadora>.

## 20. Modelo Entidad Relación

En el siguiente diagrama se representa la estructura de la base de datos, en la cual se definen las entidades principales, los atributos y las relaciones que permiten almacenar y gestionar toda la información relacionada con los usuarios del sistema, roles, dispositivos, alertas, notificaciones, historial y tips.



• **Figura 18 Modelo Entidad Relación**

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**  
**21. Diagrama de clases**

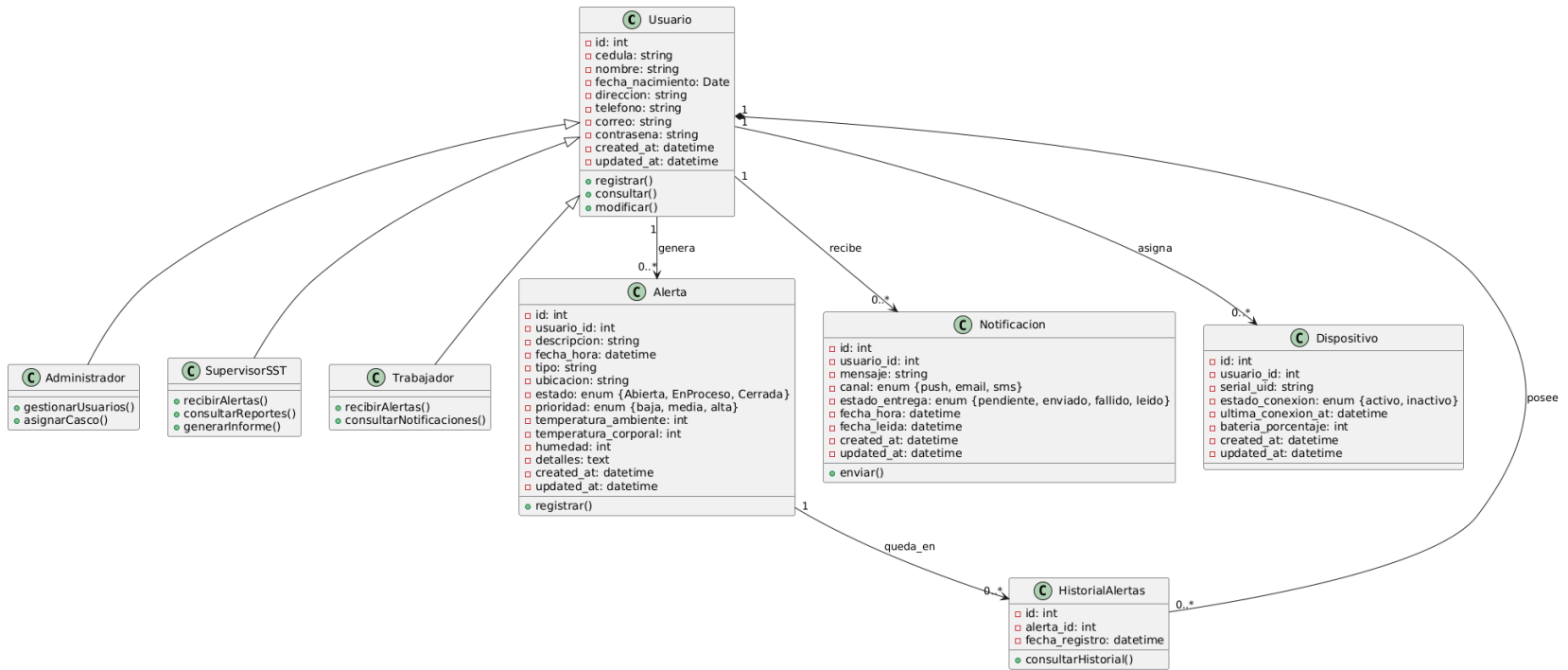
En este diagrama se ilustra toda la estructura lógica del sistema y se refleja el diseño orientado a objetos, se representan las clases principales, atributos, métodos y las relaciones de herencia que hay entre ellas, lo cual permite identificar las responsabilidades correspondientes a cada rol del sistema, así como la interacción con las alertas notificaciones y historial de alertas permitiendo reflejar el comportamiento esperado en la aplicación.

Nota aclaratoria (MER – Diagrama de clases)

En los diagramas presentados se aclara que dentro de sus entidades, clases y atributos existe información autogestionada. Este tipo de información no es gestionada manualmente por los usuarios, sino que se genera automáticamente a partir de los dispositivos y sensores (temperatura, humedad, ubicación GPS). Asimismo, se incluyen datos previamente definidos en el código, como los tips, los rangos de temperatura que activan las alertas, el manejo de roles, los informes y el historial de alertas. Toda esta información es procesada internamente por el sistema como parte de su funcionamiento y verificación de procesos, teniendo en cuenta que se trata de un prototipo.

## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

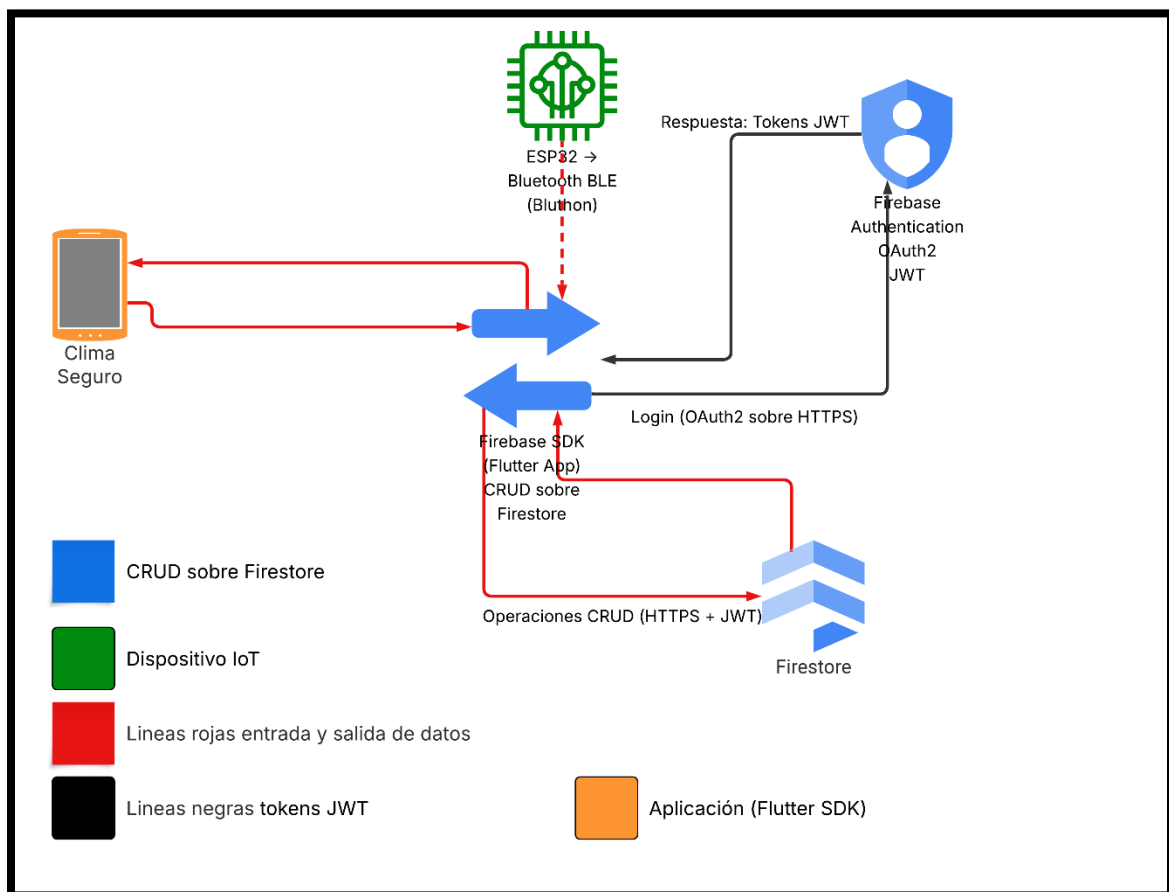


• **Figura 19 Diagrama de clases**

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**

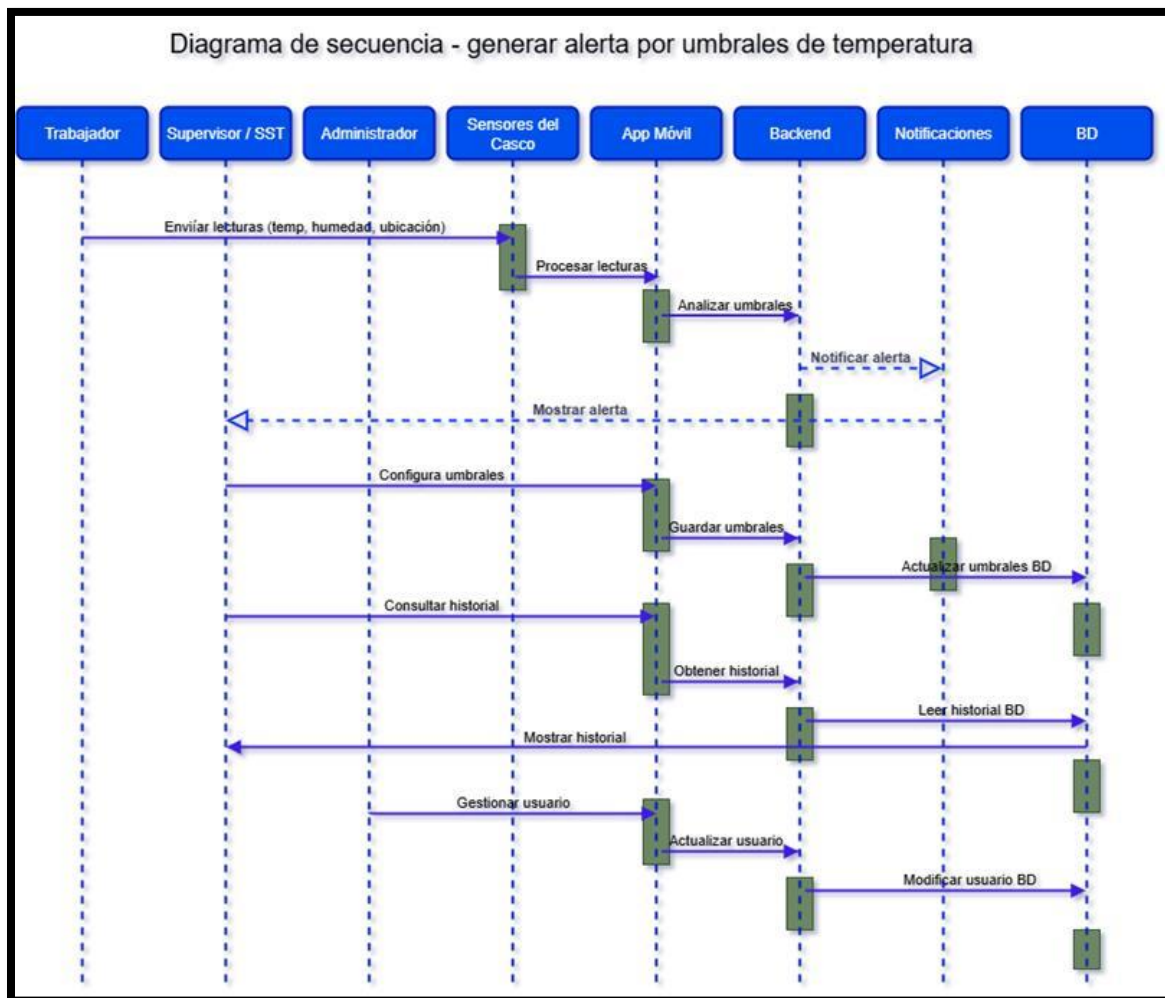
**22. Diagrama de Arquitectura**

En este diagrama mostraremos el funcionamiento del sistema del casco con el módulo ESP32, el cual recolecta información mediante los sensores de temperatura ambiente, humedad y el GPS, utilizando Bluetooth esta información será enviada hacia la aplicación, los datos serán procesados y enviados a través del protocolo HTTPS a Firebase, con Firebase Authentication, se protege el acceso a la aplicación, mediante un token JWT.



• **Figura 20 Diagrama de arquitectura**

## 23 Diagrama de Secuencia



- Figura 21 Diagrama de secuencia

## 24. Justificación de Hardware y Software

Se eligió trabajar con Flutter y Dart para desarrollar la aplicación móvil. La razón principal es que permite programar una vez y el código sirve para Android y iOS, lo cual ahorra mucho tiempo, Firebase se usó para manejar todo lo relacionado con datos. Tiene funciones útiles como almacenamiento en la nube, control de usuarios y envío de alertas, que son necesarias para que el casco se comunice con la app en tiempo real. Para la parte electrónica se escogió el ESP32 porque consume poca energía (importante para un dispositivo portable), trae Wifi y Bluetooth, y no hay problema para conectarle los sensores.

Los sensores se escogieron considerando su precisión, velocidad de respuesta, confiabilidad y que fueran accesibles en precio.

Sensor SHT31: Mide la temperatura y humedad del entorno. Se eligió porque es muy preciso ( $\pm 0.3$  °C y  $\pm 2\%$  de humedad), da lecturas rápidas y es estable en el tiempo (Naylamp Mechatronics, 2023).

Sensor MLX90614: Registra la temperatura corporal sin tocar a la persona, mediante tecnología infrarroja que responde en milisegundos. Esto lo hace seguro e higiénico, con un error de menos de  $\pm 0.5$  °C (Naylamp Mechatronics, 2023).

Módulo GPS NEO-6M: Permite rastrear dónde están los trabajadores en tiempo real, con precisión de 2.5 metros. Gasta poca batería y es más económico que otros GPS profesionales, funcionando bien para seguridad laboral. (Sigmaelectronica, 2025)

## 25. Diseño

Para esta fase se muestra el diseño de la aplicación a través de los mockups. Aquí se puede ver la estructura de las pantallas, la forma en que están organizados los elementos y la manera en que el usuario navegará por el sistema.

Las interfaces fueron diseñadas con el objetivo de ofrecer al administrador y a los trabajadores una forma sencilla de visualizar la información crítica. Cada vista está enfocada en mostrar indicadores claros, alertas oportunas y opciones de configuración intuitivas para mejorar la experiencia del usuario y garantizar la seguridad del personal.

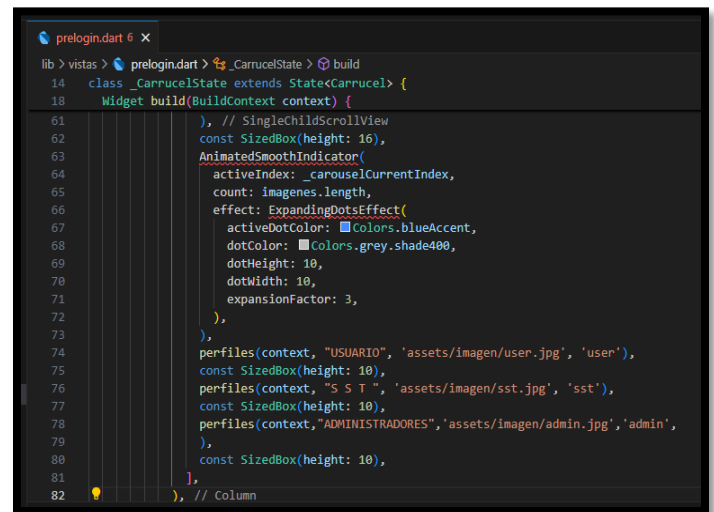
## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

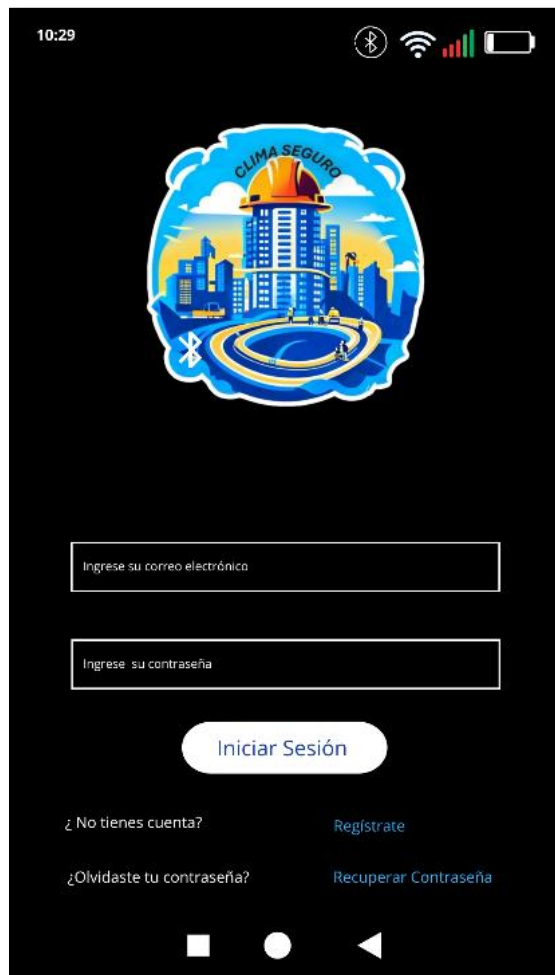
#### 25.1 Mockups



- **Figura 22 Pantalla de inicio (Usuario)**



- **Figura 23 Pantalla de inicio (Usuario)**



- **Figura 24 Login (Usuario)**

```
login.dart x
lib > vistas > login.dart > _LoginState > build
12 class _LoginState extends State<Login> {
19   widget build(BuildContext context) {
74     TextFormField(
43       style: TextStyle(color: Colors.white),
44       validator: (value) {
45         if (value == null || value.isEmpty) {
46           return 'Por favor ingrese su correo electrónico';
47         }
48         return null;
49       },
50     ),
51     keyboardType: TextInputType.emailAddress,
52     controller: _correo,
53     decoration: InputDecoration(
54       label: Text(
55         "Ingrese su correo electrónico",
56         style: TextStyle(color: Colors.white),
57       ), // Text
58       border: OutlineInputBorder(),
59     ), // InputDecoration
60   ), // TextFormField
61   SizedBox(height: 30),
62   TextFormField(
63     style: TextStyle(color: Colors.white),
64     controller: _contrasena,
65     validator: (value) {
66       if (value == null || value.isEmpty) {
67         return 'Por favor ingrese su contraseña';
68       }
69       return null;
70     },
71     obscureText: _oculto,
72     decoration: InputDecoration(
```

- **Figura 25 Login (Usuario)**

- **Figura 26 Pantalla registro nuevo usuario**

```

registro.dart 2 X
lib > vistas > registro.dart > ...
15 class _RegistroState extends State<Registro> {
28   Widget build(BuildContext context) {
32     centerTitle: true,
33     title: Text("Registro nuevo usuario"), // AppBar
34     body: SingleChildScrollView(
35       child: Container(
36         child: Form(
37           key: _formKey,
38           child: Padding(
39             padding: EdgeInsets.all(15),
40             child: Column(
41               mainAxisAlignment: MainAxisAlignment.center,
42               children: [
43                 Row(
44                   children: [
45                     Expanded(
46                       child: Datos(controller: _nombre, label: "ingrese su nombre"),
47                     ), // Expanded
48                     SizedBox(width: 10),
49                     Expanded(
50                       child: Datos(controller: _apellidos, label: "ingrese sus apellidos"),
51                     ), // Expanded
52                   ],
53                 ), // Row
54                 SizedBox(height: 20),
55                 Row(
56                   children: [
57                     Expanded(
58                       child: Datos(controller: _cedula, label: "ingrese su cedula"),
59                     ), // Expanded
60                     SizedBox(width: 10),
61                     Expanded(

```

- **Figura 27 Pantalla registro nuevo usuario**



- **Figura 28** Pantalla de recuperación contraseña (Usuario)

```

RecuperarC.dart X
10 > vista > RecuperarC.dart > _RecuperarCState @ build
11 class _RecuperarCState extends State<RecuperarC> {
12   Widget build(BuildContext context) {
13     return Scaffold(
14       appBar: AppBar(
15         title: Text("Recuperar contraseña"),
16       ), // AppBar
17       body: Container(
18         child: Form(
19           key: _formKey,
20           child: Padding(
21             padding: EdgeInsets.all(15),
22             child: Column(
23               mainAxisAlignment: MainAxisAlignment.center,
24               children: [
25                 TextFormField(
26                   keyboardType: TextInputType.emailAddress,
27                   controller: correo,
28                   validator: (value) => value!.isEmpty
29                     ? "El correo no puede estar vacío"
30                     : null,
31                   decoration: InputDecoration(
32                     label: Text("Ingrese su correo electronico", style: TextStyle(color: Colors.white, fontS
33                     ), // TextFormField
34                     border: OutlineInputBorder(), // InputDecoration
35                     // TextFormField
36                     SizedBox(height: 20),
37                     ElevatedButton (
38                       onPressed: ()async{
39                         if (_formKey.currentState!.validate()) {
40                           mensaje(
41                             await registrar().RecuperarContraseña(correo.text));
42                         }
43                       },
44                       child: Text(

```

- **Figura 29** Pantalla de recuperación contraseña (Usuario)



- **Figura 30** Pantalla confirmación envío de correo (Usuario)

```

lib > vistas > RecuperarC.dart > RecuperarCState > build
11 class _RecuperarCState extends State<RecuperarC> {
12   Widget build(BuildContext context) {
13     return Scaffold(
14       appBar: AppBar(
15         title: Text("Recuperar contraseña"),
16       ),
17       body: Column(
18         children: [
19           Text("Ingrese su correo electronico", style: TextStyle(color: Colors.white, fontS:
20             16)),
21           TextFormField(
22             decoration: InputDecoration(
23               border: OutlineInputBorder(), // InputDecoration
24             ),
25           ),
26           SizedBox(height: 20),
27           ElevatedButton(
28             onPressed: () async {
29               if (_formKey.currentState!.validate()) {
30                 mensaje(
31                   await Registrar().RecuperarContraseña(correo.text));
32               }
33             },
34             child: Text(
35               "Recuperar contraseña",
36               style: TextStyle(color: Colors.blueAccent),
37             ), // Text
38             style: ElevatedButton.stylefrom(
39               backgroundColor:
40                 Colors.white,
41               padding:
42                 EdgeInsets.symmetric(horizontal: 50, vertical: 20),
43             ), // ElevatedButton
44           ), // Column
45         ], // Padding
46       ), // Form
47     ), // Container
48   ); // Scaffold
49 }
50 void mensaje(msg) {

```

- **Figura 31** Pantalla confirmación envío de correo (Usuario)



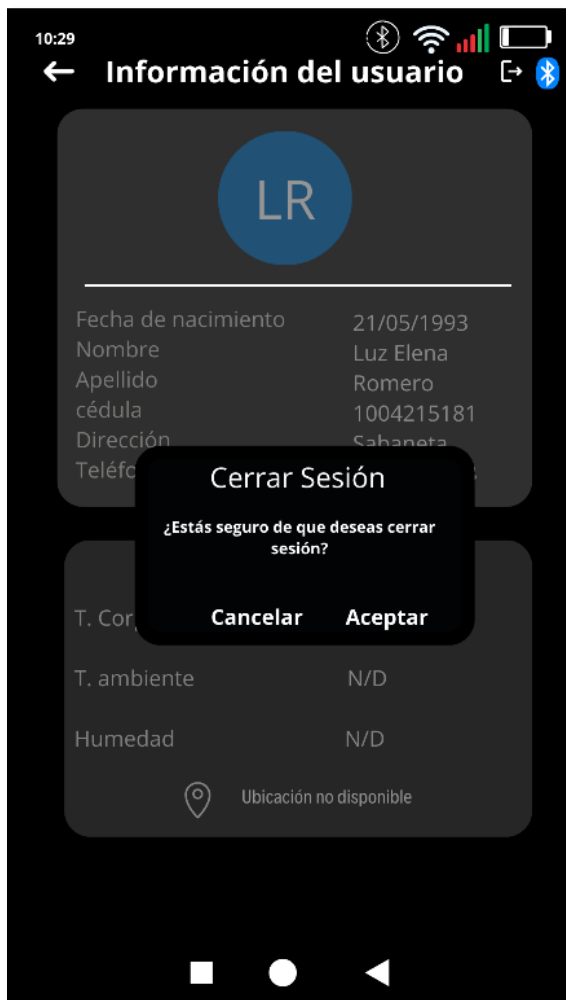
- **Figura 32 Pantalla información usuario**

```

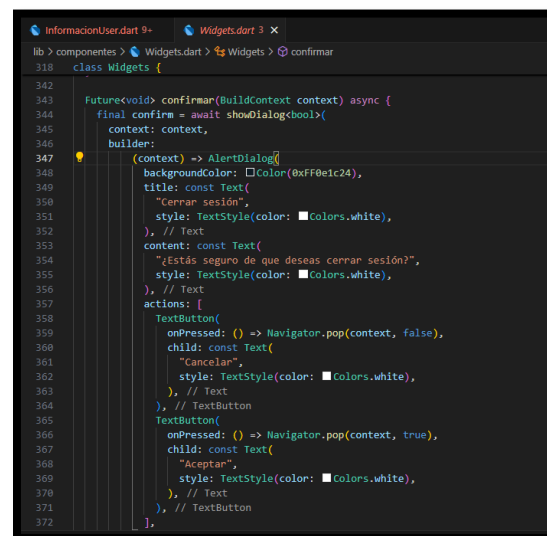
informacionUser.dart 9- x
lib > vistas > informacionUser.dart > _InformacionUserState > _buildProfileCard
16 class _InformacionUserState extends State<InformacionUser> {
193 Widget _buildProfileCard(Usuario usuario) {
195   color: colores(),
196   shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(16)),
197   child: DefaultTextStyle(
198     style: const TextStyle(color: Colors.white),
199     child: Padding(
200       padding: const EdgeInsets.all(15),
201       child: Column(
202         children: [
203           CircleAvatar(
204             radius: 50,
205             backgroundColor: Colors.blueAccent,
206             child: Text(
207               "${usuario.nombre.substring(0, 1)}${usuario.apellidos.substring(0, 1)}",
208               style: const TextStyle(
209                 fontSize: 50,
210                 color: Colors.white,
211                 fontWeight: FontWeight.bold,
212               ), // TextStyle
213             ), // Text
214           ), // CircleAvatar
215           const Divider(color: Colors.white54,
216             const SizedBox(height: 10),
217           _infoRow("Fecha de nacimiento", usuario.fechaNac),
218           _infoRow("Nombre", usuario.nombre),
219           _infoRow("Apellidos", usuario.apellidos),
220           _infoRow("Cédula", usuario.cedula),
221           _infoRow("Dirección", usuario.direccion),
222           _infoRow("Teléfono", usuario.telefono),
223         ], // Column
224       ), // Column

```

- **Figura 33 Pantalla información usuario**



- **Figura 34 Pantalla cerrar sesión (Usuario)**



- **Figura 35 Pantalla cerrar sesión (Usuario)**



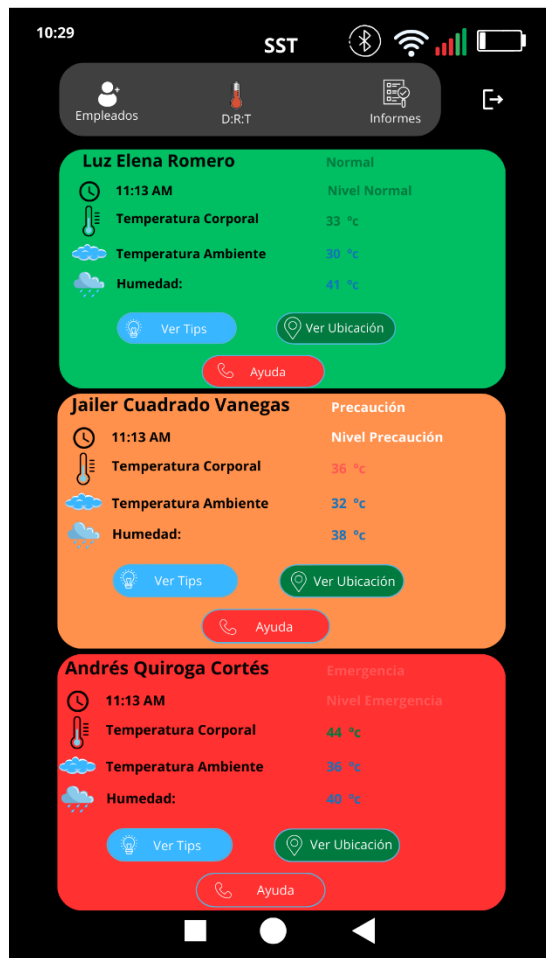
- Figura 36 Pantalla de inicio (SST)

```

prelogin.dart 6 x
lib > vistas > prelogin.dart > _CarrucelState > build
14 class _CarrucelState extends State<Carrucel> {
18   Widget build(BuildContext context) {
61     ), // SingleChildScrollView
62     const SizedBox(height: 16),
63     AnimatedSmoothIndicator(
64       activeIndex: _carouselCurrentIndex,
65       count: imagenes.length,
66       effect: ExpandingDotsEffect(
67         activeDotColor: Colors.blueAccent,
68         dotColor: Colors.grey.shade400,
69         dotHeight: 10,
70         dotWidth: 10,
71         expansionFactor: 3,
72       ),
73     ),
74     perfiles(context, "USUARIO", 'assets/imagen/user.jpg', 'user'),
75     const SizedBox(height: 10),
76     perfiles(context, "S S T", 'assets/imagen/sst.jpg', 'sst'),
77     const SizedBox(height: 10),
78     perfiles(context, "ADMINISTRADORES", 'assets/imagen/admin.jpg', 'admin',
79     ),
80     const SizedBox(height: 10),
81   ],
82   ), // Column

```

- Figura 37 Pantalla de inicio (SST)



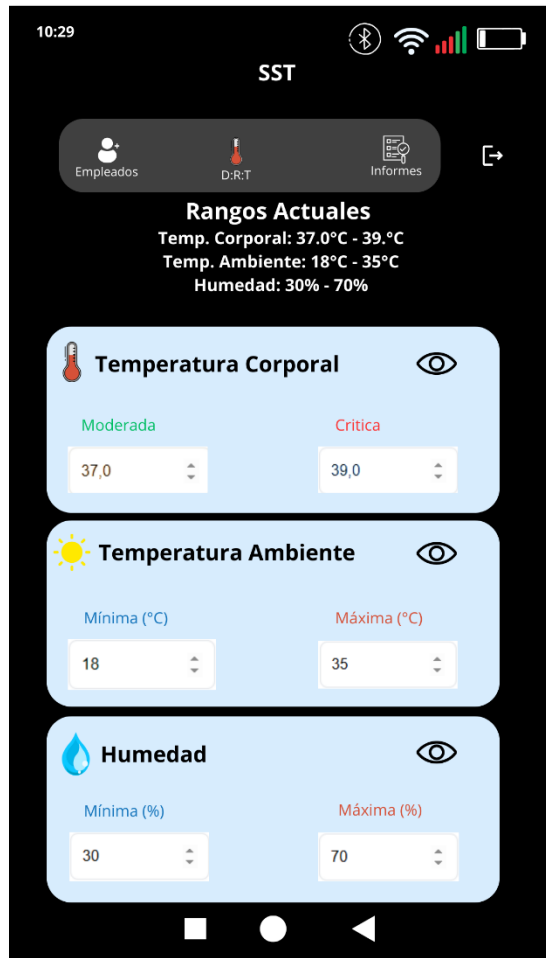
- **Figura 38** Pantalla vista monitoreo empleados (SST)

```

clima_seguro > clima_seguro > lib > vistas > SstVista.dart > ...
1  import 'package:clima_seguro/componentes/Componentes.dart';
2  import 'package:clima_seguro/model/entity/Usuario.dart';
3  import 'package:clima_seguro/model/repository/UsuariosRepo.dart';
4  import 'package:flutter/material.dart';
5  import 'package:font_awesome_flutter/font_awesome_flutter.dart';
6  import 'package:intl/intl.dart';
7  import 'package:url_launcher/url_launcher.dart';
8
9  class Sstvista extends StatefulWidget {
10     const Sstvista({super.key});
11
12     @override
13     State<Sstvista> createState() => _SstvistaState();
14 }
15
16 class _SstvistaState extends State<Sstvista> {
17     final Usuariosrepo _repositorio = Usuariosrepo();
18     final AlarmPlayer _alarmPlayer = AlarmPlayer();
19     void _llamarEmergencia(String numero) async {
20         final Uri url = Uri(scheme: numero, path: numero);
21         if (await canLaunchUrl(url)) {
22             await launchUrl(url);
23         } else {
24             ScaffoldMessenger.of(context).showSnackBar(
25                 const SnackBar(content: Text('No se pudo realizar la llamada')),
26             );
27         }
28     }
29
30     @override
31     Widget build(BuildContext context) {
32         return Scaffold(

```

- **Figura 39** Pantalla vista monitoreo empleados (SST)



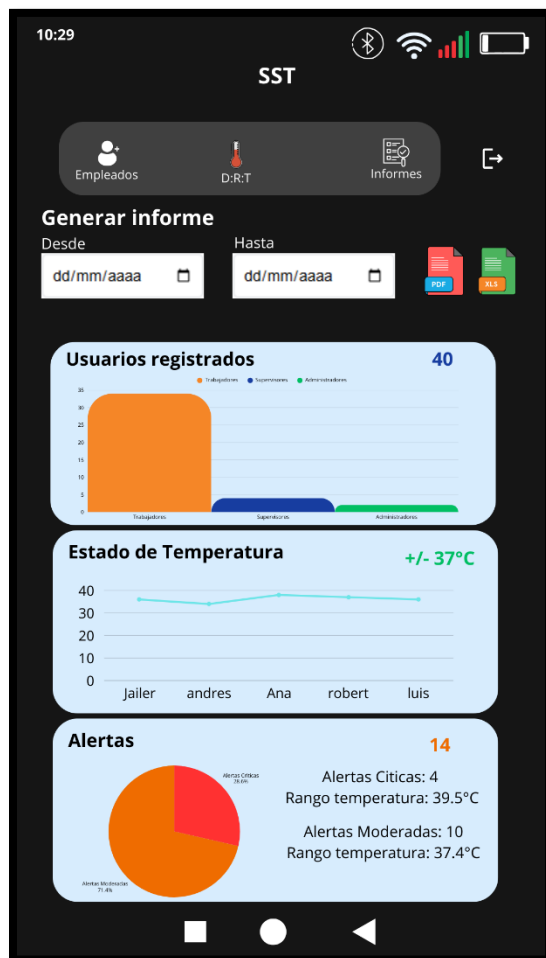
- **Figura 40 Pantalla rangos de temperatura (SST)**

```

class RangosConfigState extends State<RangosConfig> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Rangos Actuales"),
        backgroundColor: Colors.blueAccent,
      ), // AppBar
      body: SingleChildScrollView(
        padding: const EdgeInsets.all(10),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            const Text("Temp. Corporal: 37.0°C - 39.0°C"),
            const Text("Temp. Ambiente: 18°C - 35°C"),
            const Text("Humedad: 30% - 70%"),
            const SizedBox(height: 20),
            // Temperatura corporal
            rangocard(
              titulo: "Temperatura Corporal",
              icono: Icons.thermostat,
              color: Colors.redAccent,
              min: "37.0",
              max: "39.0",
            ),
            const SizedBox(height: 20),
            // Temperatura ambiente
            rangocard(
              titulo: "Temperatura Ambiente",

```

- **Figura 41 Pantalla rangos de temperatura (SST)**



- **Figura 42** Pantalla informes y consultas (SST)

```

class _DashboardAdminState extends State<DashboardAdmin> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Sistemas de emergencias - Admin"),
        backgroundColor: colors.blueAccent,
      ), // AppBar
      body: SingleChildScrollView(
        padding: const EdgeInsets.all(16),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            // Generar Informe
            Row(
              children: [
                Expanded(
                  child: TextField(
                    decoration: InputDecoration(
                      labelText: "Desde",
                      hintText: "dd/mm/aaaa",
                      filled: true,
                      fillColor: colors.white,
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(12),
                      ), // OutlineInputBorder
                    ), // InputDecoration
                  ), // TextField
                Expanded(
                  child: TextField(

```

- **Figura 43** Pantalla informes y consultas (SST)



- **Figura 44 pantalla de inicio (Administrador)**

```

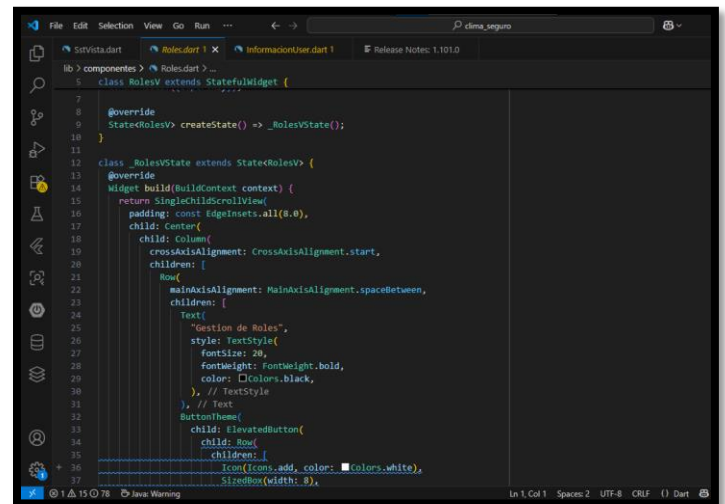
prelogin.dart 6 x
lib > vistas > prelogin.dart > _CarruselState > build
14 class _CarruselState extends State<Carrusel> {
18   widget build(BuildContext context) {
61     ), // SingleChildScrollView
62     const SizedBox(height: 16),
63     AnimatedSmoothIndicator(
64       activeIndex: _carouselCurrentIndex,
65       count: imagenes.length,
66       effect: ExpandingDotsEffect(
67         activeDotColor: Colors.blueAccent,
68         dotColor: Colors.grey.shade400,
69         dotHeight: 10,
70         dotWidth: 10,
71         expansionFactor: 3,
72       ),
73     ),
74     perfiles(context, "USUARIO", 'assets/imagen/user.jpg', 'user'),
75     const SizedBox(height: 10),
76     perfiles(context, "S S T", 'assets/imagen/sst.jpg', 'sst'),
77     const SizedBox(height: 10),
78     perfiles(context, "ADMINISTRADORES", 'assets/imagen/admin.jpg', 'admin',
79     ),
80     const SizedBox(height: 10),
81   ],
82   ), // Column

```

- **Figura 45 pantalla de inicio (Administrador)**



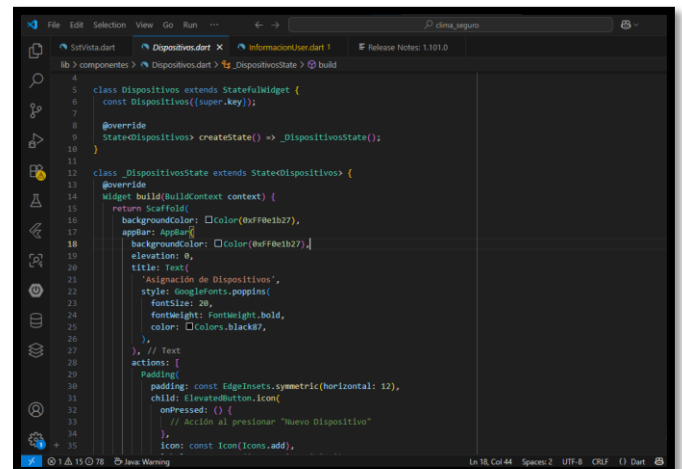
- **Figura 46 Pantalla Gestión de roles (Admin)**



- **Figura 47 Pantalla Gestión de roles (Admin)**



- **Figura 48 Pantalla asignación dispositivo (Admin)**



- **Figura 49 Pantalla asignación dispositivo (Admin)**



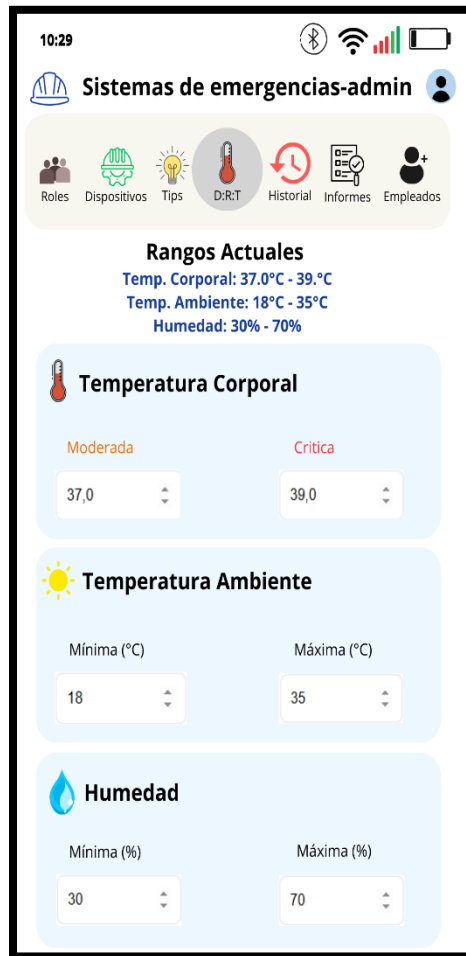
- **Figura 50 Pantalla gestión de tips (Admin)**

```

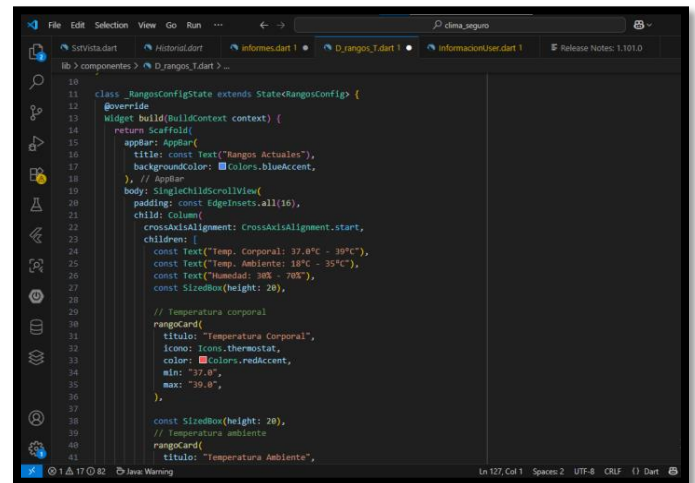
1  class TipsState extends StatefulWidget {
2    @override
3    createState() => _TipsState();
4  }
5
6  class _TipsState extends State<Tips> {
7    @override
8    Widget build(BuildContext context) {
9      return Scaffold(
10         backgroundColor: Color(0xFF013277),
11         appBar: AppBar(
12           backgroundColor: Color(0xFF013277),
13           title: Text(
14             "Gestión de Tips",
15             style: TextStyle(
16               fontSize: 24,
17               fontWeight: FontWeight.bold,
18               color: Colors.black87,
19             ), // TextStyle
20           // TextStyle
21         ), // TextStyle
22         actions: [
23           Padding(
24             padding: const EdgeInsets.symmetric(horizontal: 12),
25             child: ElevatedButton.icon(
26               onPressed: () {},
27               icon: Icon(Icons.add),
28               label: const Text("Nuevo Tip"),
29               style: ElevatedButton.styleFrom(
30                 backgroundColor: const Color(0xFF2579FF),
31                 foregroundColor: Colors.white,
32               ),
33             ),
34           ],
35         ),
36       );
37     }
38   }

```

- **Figura 51 Pantalla gestión de tips (Admin)**



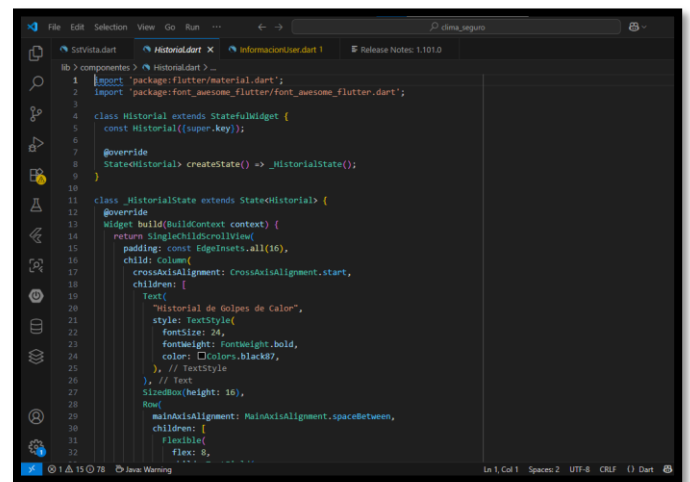
- **Figura 52 Pantalla rangos de temperatura (Admin)**



- **Figura 53 Pantalla rangos de temperatura (Admin)**



- **Figura 54** Pantalla historial golpes de calor (Admin)



- **Figura 55** Pantalla historial golpes de calor (Admin)



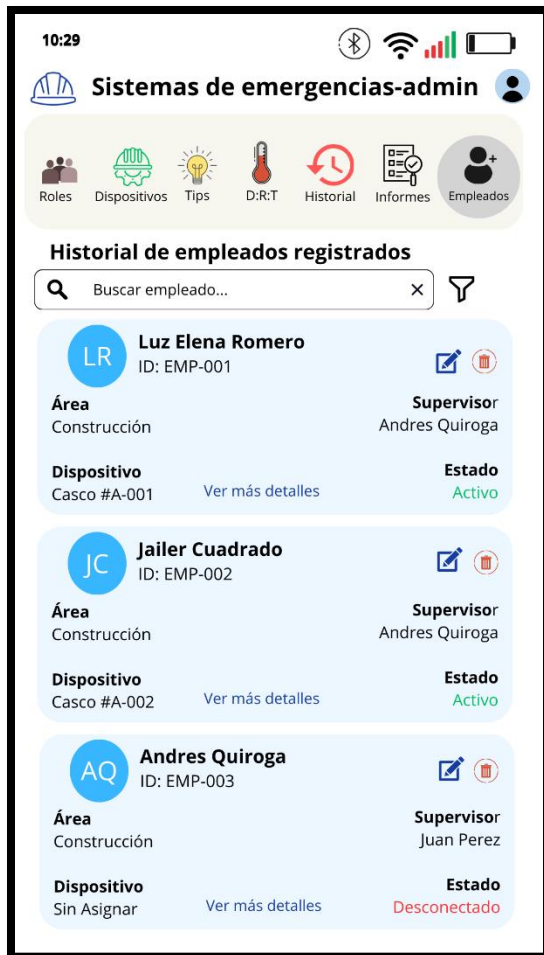
- **Figura 56 Pantalla de informes y consultas (Admin)**

```

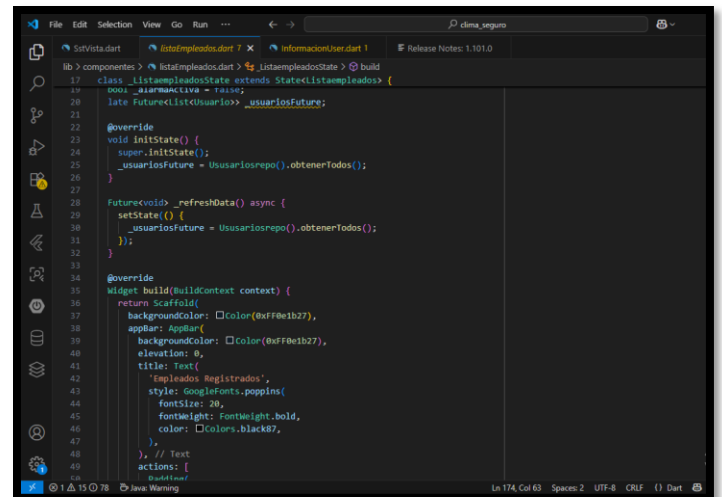
class DashboardAdminState extends State<DashboardAdmin> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Sistemas de emergencias - Admin"),
        backgroundColor: Colors.blueAccent,
      ), // AppBar
      body: SingleChildScrollView(
        padding: const EdgeInsets.all(16),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            // Generar Informe
            Row(
              children: [
                Expanded(
                  child: TextField(
                    decoration: InputDecoration(
                      labelText: "Desde",
                      hintText: "dd/mm/aaaa",
                      filled: true,
                      fillColor: Colors.white,
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(12),
                      ), // OutlineInputBorder
                    ), // InputDecoration
                  ), // TextField
                Expanded(
                  child: TextField(
                    decoration: InputDecoration(
                      labelText: "Hasta",
                      hintText: "dd/mm/aaaa",
                      filled: true,
                      fillColor: Colors.white,
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(12),
                      ), // OutlineInputBorder
                    ), // InputDecoration
                  ), // TextField
                const SizedBox(width: 16),
                Expanded(
                  child: TextField(
                    decoration: InputDecoration(
                      labelText: "PDF",
                      filled: true,
                      fillColor: Colors.white,
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(12),
                      ), // OutlineInputBorder
                    ), // InputDecoration
                  ), // TextField
                Expanded(
                  child: TextField(
                    decoration: InputDecoration(
                      labelText: "XLS",
                      filled: true,
                      fillColor: Colors.white,
                      border: OutlineInputBorder(
                        borderRadius: BorderRadius.circular(12),
                      ), // OutlineInputBorder
                    ), // InputDecoration
                  ), // TextField
            ], // Row
          ], // Column
        ), // SingleChildScrollView
      ), // Scaffold
    );
  }
}

```

- **Figura 57 Pantalla de informes y consultas (Admin)**



- **Figura 58** Pantalla empleados registrados



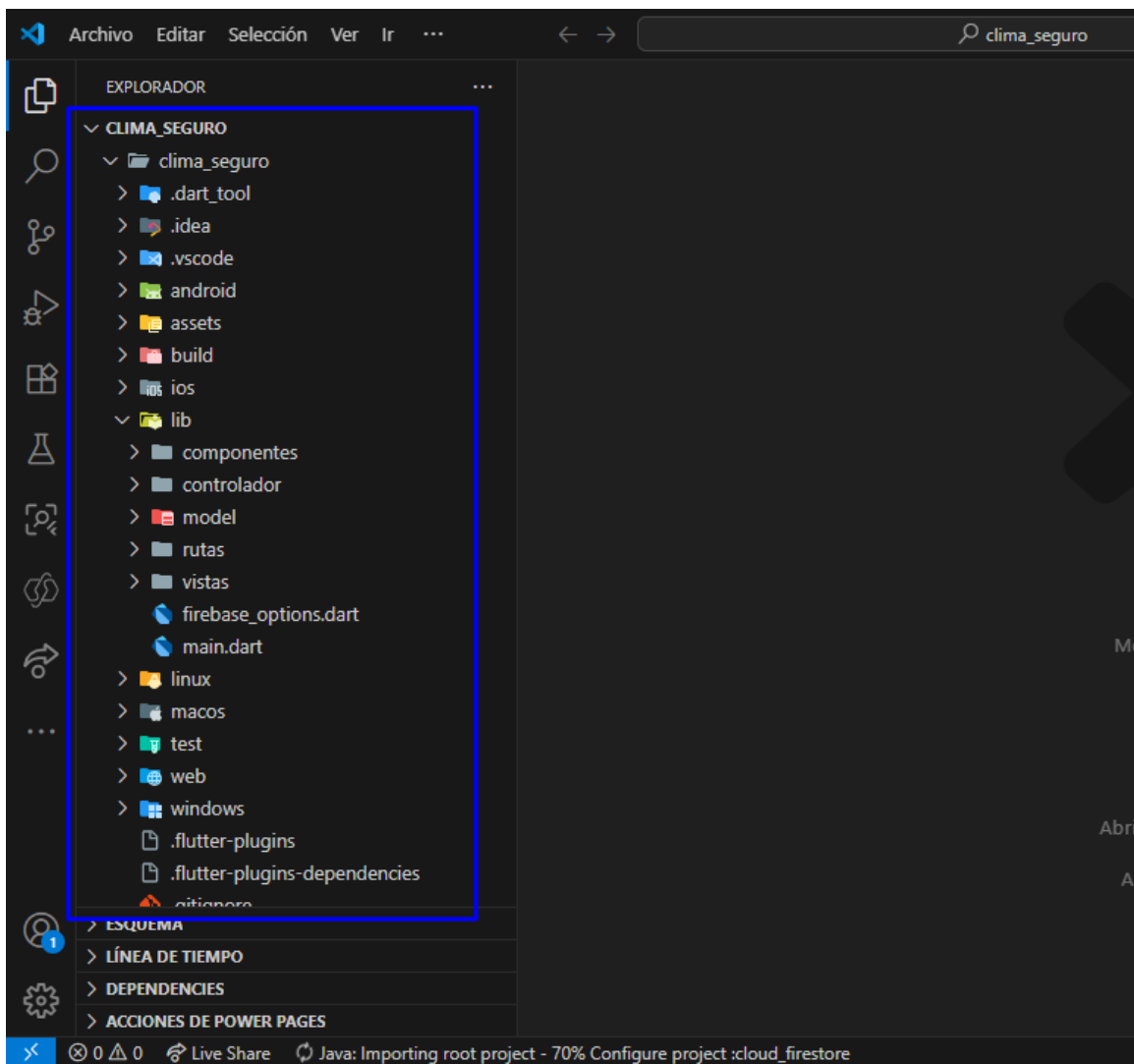
- **Figura 59** Pantalla empleados registrados

## TRABAJO DE GRADO Opción Investigación o Proyecto de Grado

### 26. Fase de Desarrollo

#### 26.1 Estructura del Proyecto

Estructura de carpetas y archivos del proyecto, en esta imagen visualizamos las carpetas **lib** aquí encontraremos el código fuente de la aplicación (modelos, vistas, controladores, rutas, componentes).



## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

- **Figura 60** Estructura del Proyecto

## 26.2 Configuración de Entornos

- **IDE usado**

El proyecto se desarrolla en **Visual Studio Code**, un entorno de desarrollo liviano y multiplataforma que permite trabajar con Flutter y Dart de manera organizada.

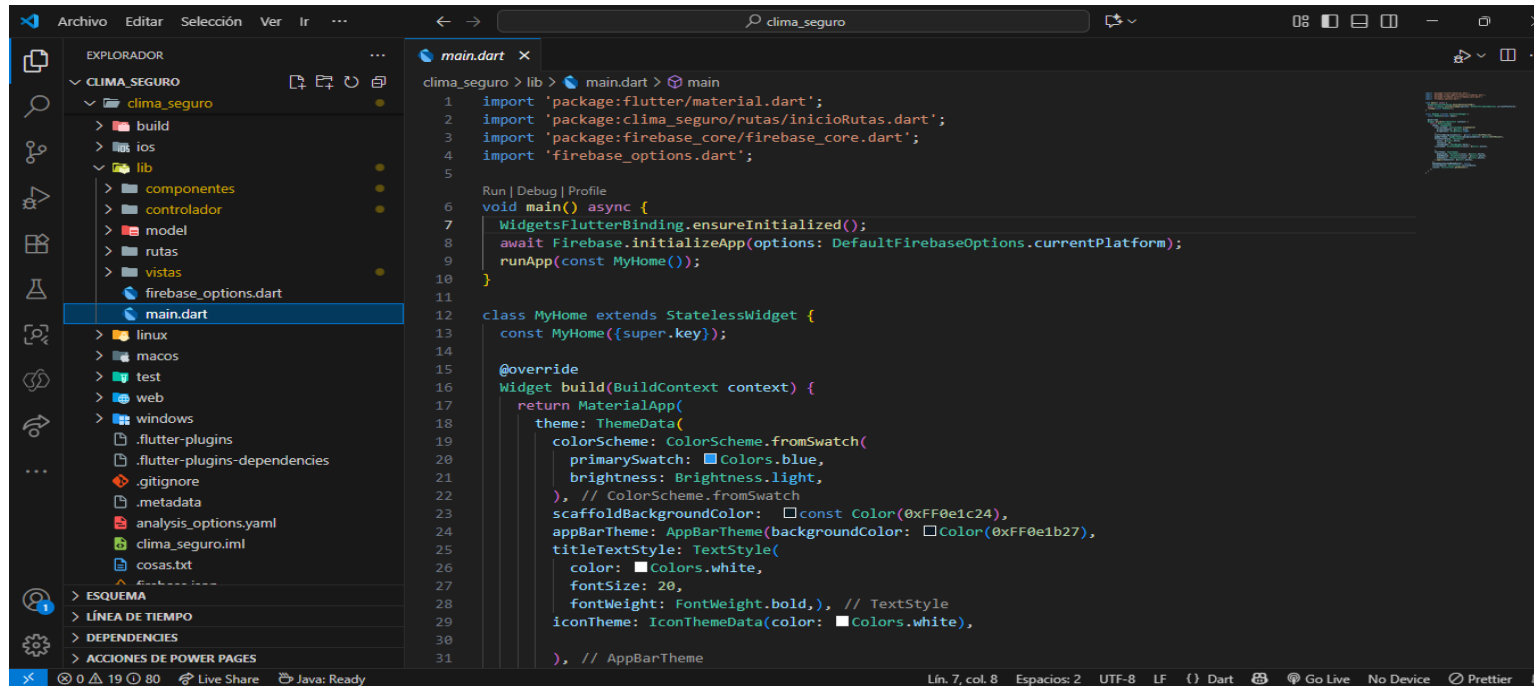
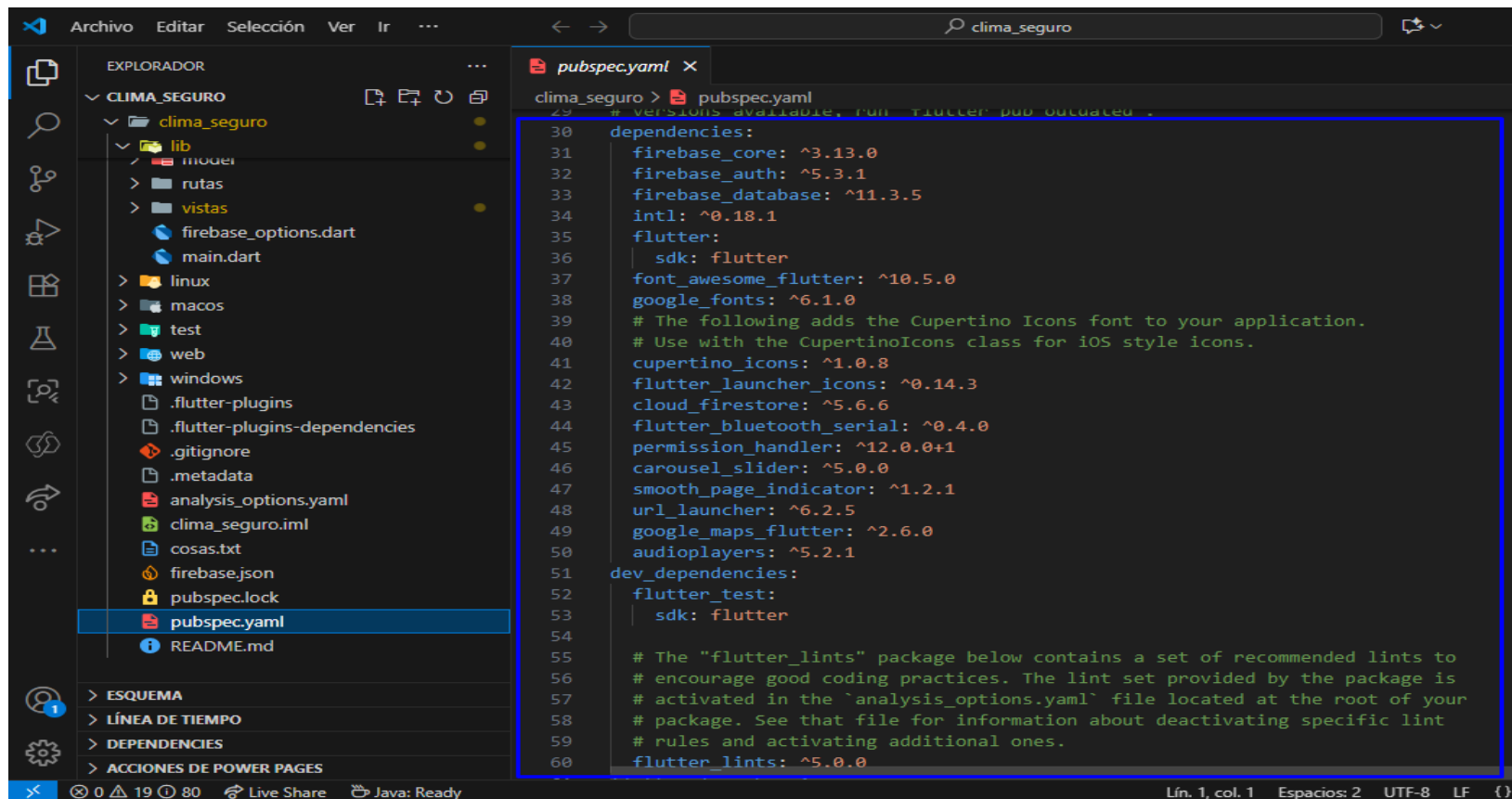


Figura 61 IDE usado

- **Dependencias instaladas** En esta carpeta pubspec.yaml se encuentran las dependencias, que son paquetes o librerías externas, como son Firebase para manejar la autenticación y la base de datos, Google Maps para mostrar la ubicación en la app, y otras librerías como audioplayers para reproducir audios y sonidos.



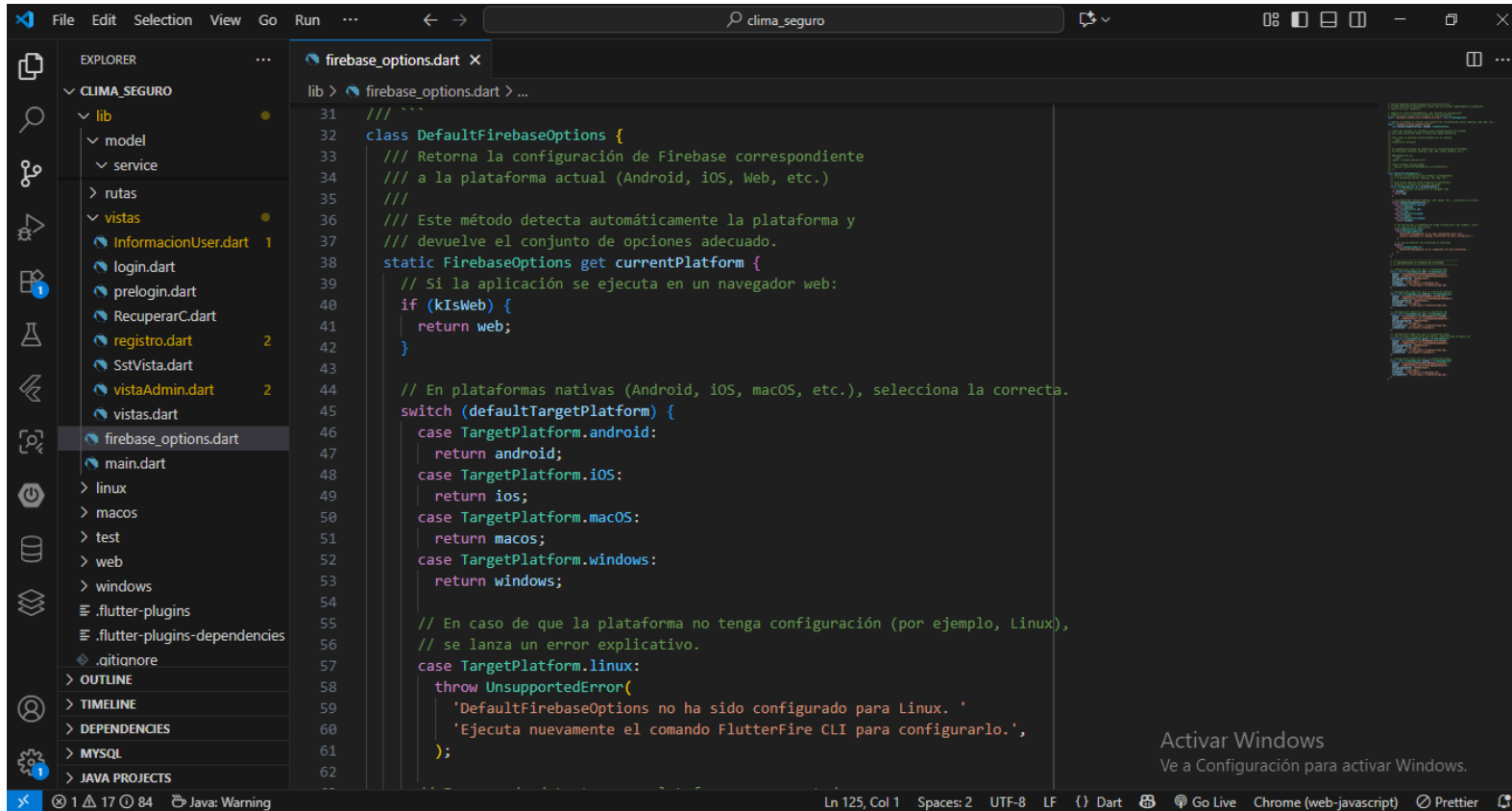
The image shows a screenshot of an IDE (Visual Studio Code) with the file explorer on the left and the editor on the right. The file explorer shows the project structure for 'CLIMA\_SEGURO', with the 'lib' folder expanded to show 'pubspec.yaml' selected. The editor displays the content of 'pubspec.yaml', which lists various dependencies for the Flutter application. The dependencies are as follows:

```
dependencies:
  firebase_core: ^3.13.0
  firebase_auth: ^5.3.1
  firebase_database: ^11.3.5
  intl: ^0.18.1
  flutter:
    sdk: flutter
  font_awesome_flutter: ^10.5.0
  google_fonts: ^6.1.0
  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.8
  flutter_launcher_icons: ^0.14.3
  cloud_firestore: ^5.6.6
  flutter_bluetooth_serial: ^0.4.0
  permission_handler: ^12.0.0+1
  carousel_slider: ^5.0.0
  smooth_page_indicator: ^1.2.1
  url_launcher: ^6.2.5
  google_maps_flutter: ^2.6.0
  audioplayers: ^5.2.1
dev_dependencies:
  flutter_test:
    sdk: flutter
  # The "flutter_lints" package below contains a set of recommended lints to
  # encourage good coding practices. The lint set provided by the package is
  # activated in the `analysis_options.yaml` file located at the root of your
  # package. See that file for information about deactivating specific lint
  # rules and activating additional ones.
  flutter_lints: ^5.0.0
```

Figura 62 Dependencias instaladas

- **Configuración De Firebase**

Este se configuró para manejar la autenticación y la base de datos en la nube. Se generó el archivo `firebase_options.dart` automáticamente al registrar la aplicación en Firebase y se agregaron los archivos de configuración (`google-services.json` para Android y `GoogleService-Info.plist` para iOS).

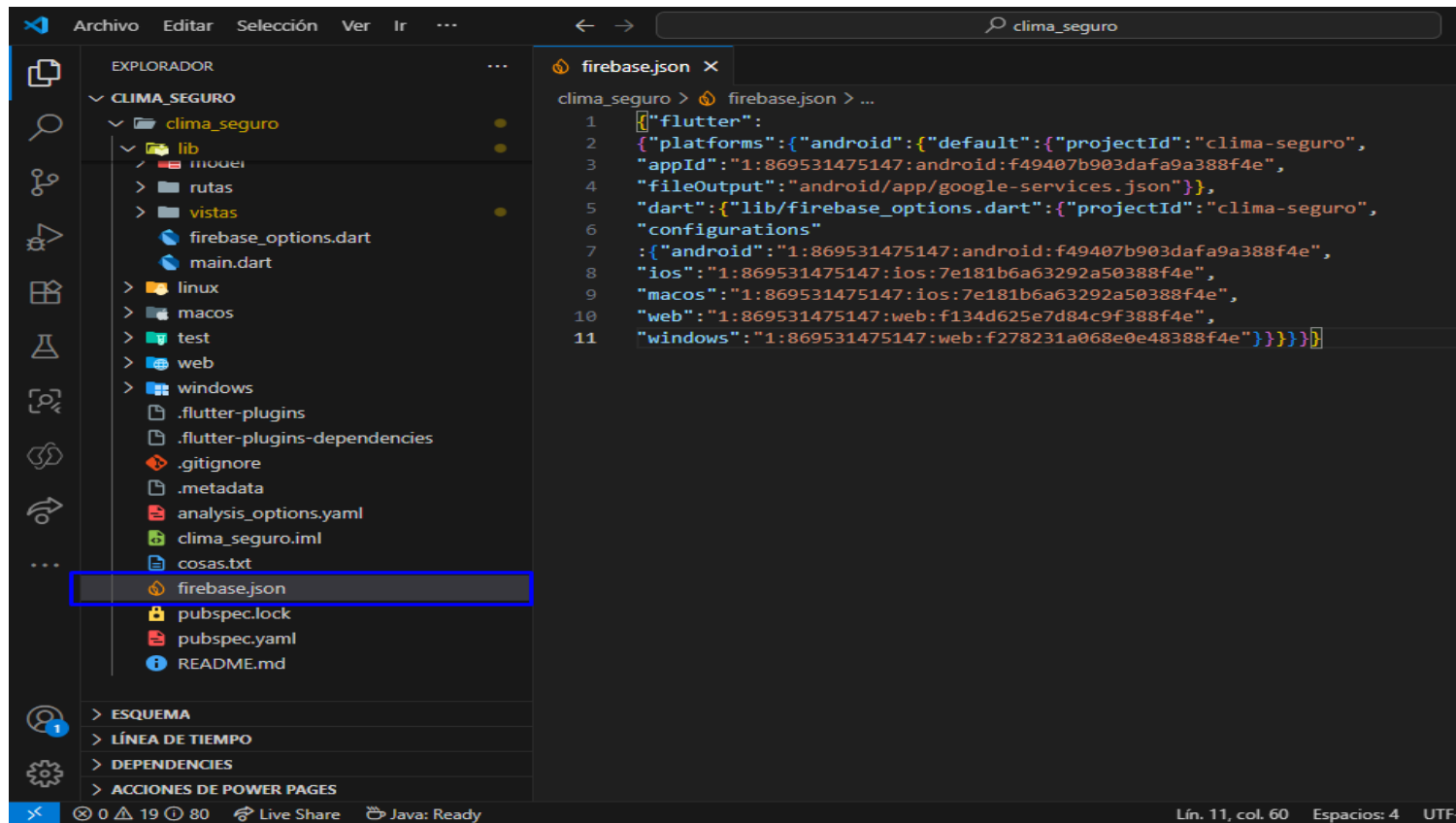


The screenshot shows an IDE window titled 'clima\_seguro' with the file 'firebase\_options.dart' open. The Explorer panel on the left shows a project structure with folders like 'lib', 'model', 'service', 'vistas', and 'main.dart'. The main editor displays the following Dart code:

```
lib > firebase_options.dart > ...
31  /// ***
32  class DefaultFirebaseOptions {
33  /// Retorna la configuración de Firebase correspondiente
34  /// a la plataforma actual (Android, iOS, Web, etc.)
35  ///
36  /// Este método detecta automáticamente la plataforma y
37  /// devuelve el conjunto de opciones adecuado.
38  static FirebaseOptions get currentPlatform {
39  // Si la aplicación se ejecuta en un navegador web:
40  if (kIsWeb) {
41  return web;
42  }
43
44  // En plataformas nativas (Android, iOS, macOS, etc.), selecciona la correcta.
45  switch (defaultTargetPlatform) {
46  case TargetPlatform.android:
47  return android;
48  case TargetPlatform.iOS:
49  return ios;
50  case TargetPlatform.macOS:
51  return macos;
52  case TargetPlatform.windows:
53  return windows;
54
55  // En caso de que la plataforma no tenga configuración (por ejemplo, Linux),
56  // se lanza un error explicativo.
57  case TargetPlatform.linux:
58  throw UnsupportedError(
59  'DefaultFirebaseOptions no ha sido configurado para Linux. '
60  'Ejecuta nuevamente el comando FlutterFire CLI para configurarlo.',
61  );
62  }
```

- **Figura 63 Configuración De Firebase**

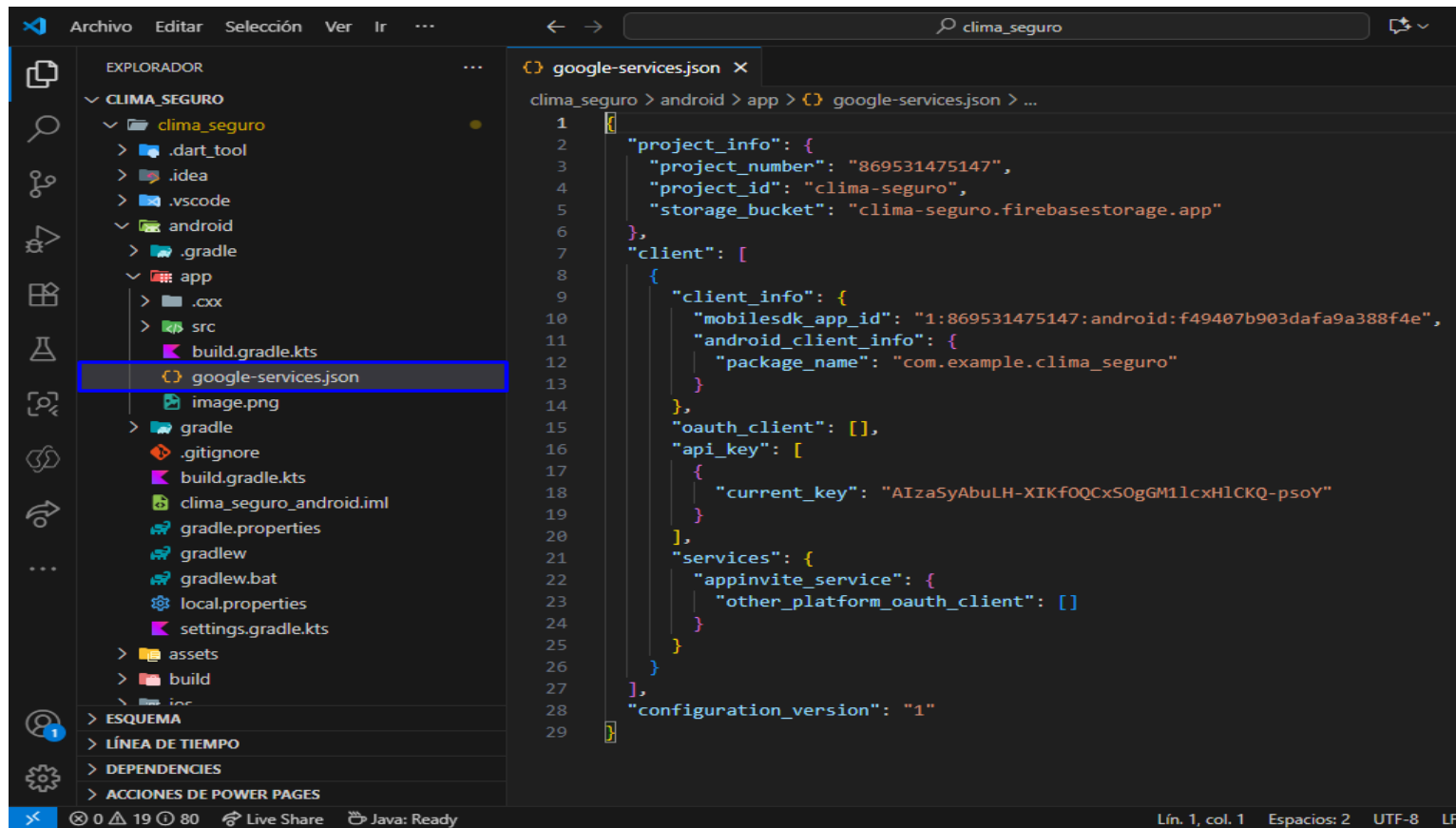
## Firebase json



```
firebase.json
1  {"flutter":
2  {"platforms":{"android":{"default":{"projectId":"clima-seguro",
3  "appId":"1:869531475147:android:f49407b903dafa9a388f4e",
4  "fileOutput":"android/app/google-services.json"}},
5  "dart":{"lib/firebase_options.dart":{"projectId":"clima-seguro",
6  "configurations"
7  :{"android":"1:869531475147:android:f49407b903dafa9a388f4e",
8  "ios":"1:869531475147:ios:7e181b6a63292a50388f4e",
9  "macos":"1:869531475147:ios:7e181b6a63292a50388f4e",
10 "web":"1:869531475147:web:f134d625e7d84c9f388f4e",
11 "windows":"1:869531475147:web:f278231a068e0e48388f4e"}}}}}}
```

- Figura 64 fairebase json

- Google Services Json



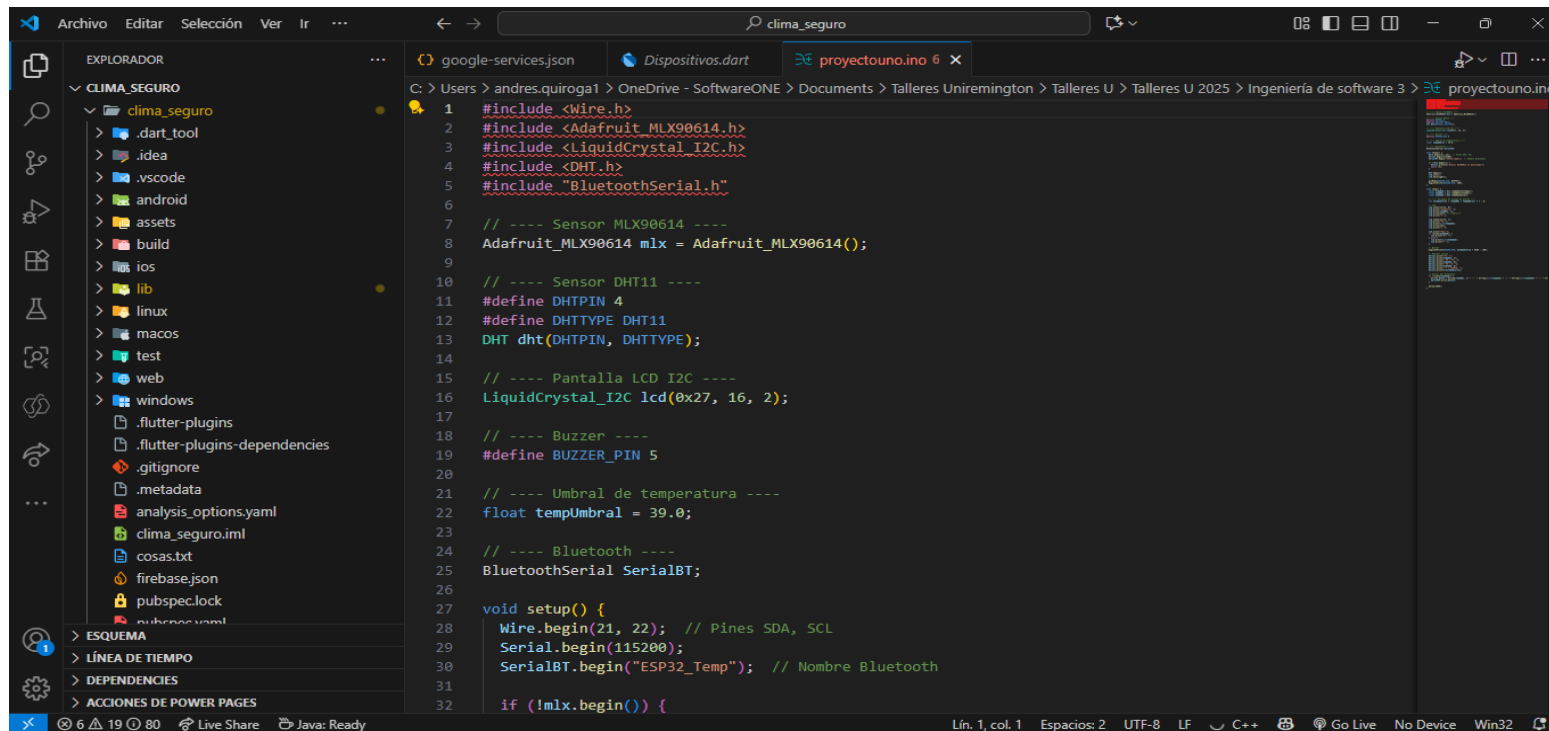
The image shows a screenshot of the Visual Studio Code editor. On the left, the Explorer sidebar shows a project structure for 'CLIMA\_SEGURO'. The file 'google-services.json' is highlighted with a blue selection bar. The main editor area displays the content of this file, which is a JSON configuration for Google services. The JSON includes project information, client information, API keys, and service configurations.

```
1  [
2  {
3    "project_info": {
4      "project_number": "869531475147",
5      "project_id": "clima-seguro",
6      "storage_bucket": "clima-seguro.firebaseioapp"
7    },
8    "client": [
9      {
10     "client_info": {
11       "mobilesdk_app_id": "1:869531475147:android:f49407b903dafa9a388f4e",
12       "android_client_info": {
13         "package_name": "com.example.clima_seguro"
14       }
15     },
16     "oauth_client": [],
17     "api_key": [
18       {
19         "current_key": "AIzaSyAbuLH-XIKf0QCxS0gGM1cxHlCKQ-psoY"
20       }
21     ],
22     "services": {
23       "appinvite_service": {
24         "other_platform_oauth_client": []
25       }
26     }
27   }
28 ],
29 "configuration_version": "1"
```

- Figura 65 google -servise jison

- **Configuración de Arduino**

En el Arduino se programó un microcontrolador (ESP32) para trabajar con diferentes sensores y módulos. Tenemos el sensor (MLX90614) que nos mide la temperatura corporal, el sensor (DHT11) para medir la temperatura y humedad ambiental, una pantalla (LCD I2C) nos muestra los valores, y un (buzzer) que se activa cuando la temperatura sobrepasa los 39 °C. se configuró la comunicación por medio de Bluetooth para enviar la información a la APP y que la aplicación pueda mostrarla en tiempo real.



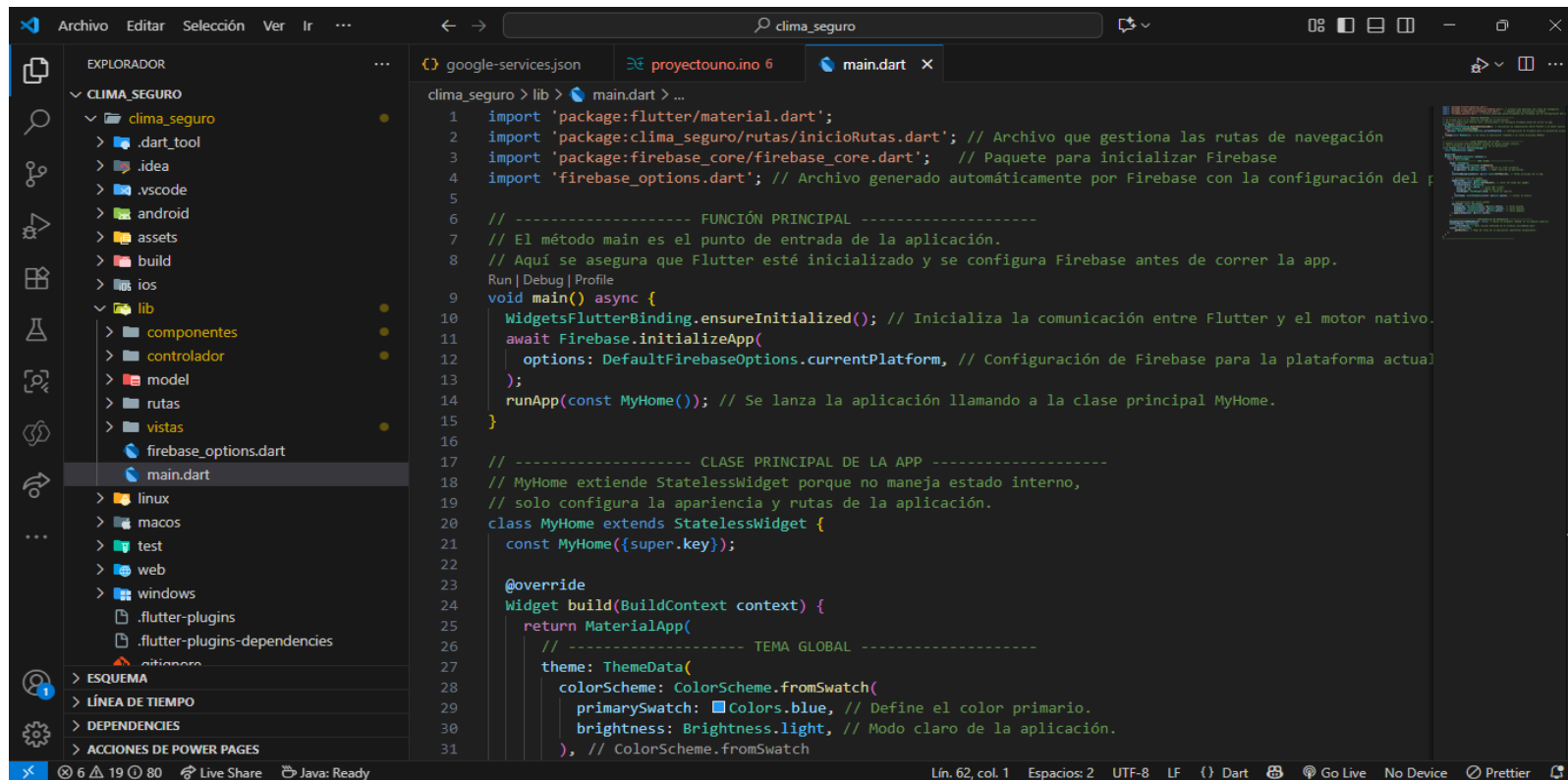
```
1 #include <Wire.h>
2 #include <Adafruit_MLX90614.h>
3 #include <LiquidCrystal_I2C.h>
4 #include <DHT.h>
5 #include "BluetoothSerial.h"
6
7 // ---- Sensor MLX90614 ----
8 Adafruit_MLX90614 mlx = Adafruit_MLX90614();
9
10 // ---- Sensor DHT11 ----
11 #define DHTPIN 4
12 #define DHTTYPE DHT11
13 DHT dht(DHTPIN, DHTTYPE);
14
15 // ---- Pantalla LCD I2C ----
16 LiquidCrystal_I2C lcd(0x27, 16, 2);
17
18 // ---- Buzzer ----
19 #define BUZZER_PIN 5
20
21 // ---- Umbral de temperatura ----
22 float tempUmbral = 39.0;
23
24 // ---- Bluetooth ----
25 BluetoothSerial SerialBT;
26
27 void setup() {
28   Wire.begin(21, 22); // Pines SDA, SCL
29   Serial.begin(115200);
30   SerialBT.begin("ESP32_Temp"); // Nombre Bluetooth
31
32   if (!mlx.begin()) {
```

- **Figura 66 Configuración de Arduino**

## 26.3 Código Fuente Relevante

- Main

Es el punto de entrada de la aplicación. Aquí se inicializa Firebase y se ejecuta la función runApp() que arranca la app.



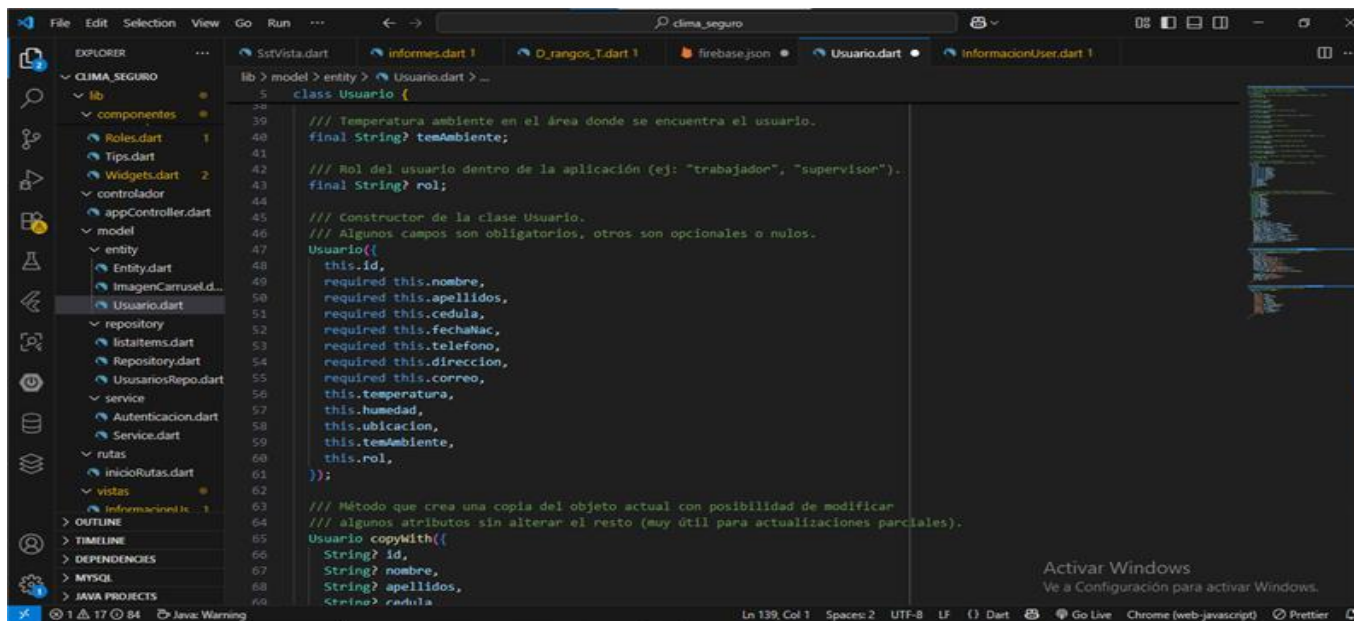
```
1 import 'package:flutter/material.dart';
2 import 'package:clima_seguro/rutas/inicioRutas.dart'; // Archivo que gestiona las rutas de navegación
3 import 'package:firebase_core/firebase_core.dart'; // Paquete para inicializar Firebase
4 import 'firebase_options.dart'; // Archivo generado automáticamente por Firebase con la configuración del p
5
6 // ----- FUNCIÓN PRINCIPAL -----
7 // El método main es el punto de entrada de la aplicación.
8 // Aquí se asegura que Flutter esté inicializado y se configura Firebase antes de correr la app.
9 void main() async {
10   WidgetsFlutterBinding.ensureInitialized(); // Inicializa la comunicación entre Flutter y el motor nativo.
11   await Firebase.initializeApp(
12     options: DefaultFirebaseOptions.currentPlatform, // Configuración de Firebase para la plataforma actual
13   );
14   runApp(const MyHome()); // Se lanza la aplicación llamando a la clase principal MyHome.
15 }
16
17 // ----- CLASE PRINCIPAL DE LA APP -----
18 // MyHome extiende StatelessWidget porque no maneja estado interno,
19 // solo configura la apariencia y rutas de la aplicación.
20 class MyHome extends StatelessWidget {
21   const MyHome({super.key});
22
23   @override
24   Widget build(BuildContext context) {
25     return MaterialApp(
26       // ----- TEMA GLOBAL -----
27       theme: ThemeData(
28         colorScheme: ColorScheme.fromSwatch(
29           primarySwatch: Colors.blue, // Define el color primario.
30           brightness: Brightness.light, // Modo claro de la aplicación.
31         ), // ColorScheme.fromSwatch
```

- **Figura 67** Main

- **Models**

### Clase usuario

El model: En este proyecto se encarga de gestionar y representar la información correspondiente a la aplicación, se definen las entidades principales junto con sus atributos y métodos para luego poder transformar los datos entre objetos y estructuras compatibles con Firebase, el model maneja los repositorios de los cuales nos permiten realizar almacenamiento de operaciones específicas, consultar y actualizar en la base de datos en tiempo real como también el manejo de autenticación de los usuarios mediante Firebase Authentication, de igual forma sirve como el puente entre la lógica y los servicios externos de esta forma se busca manejar una estructura organizada y coherente para el manejo de datos.



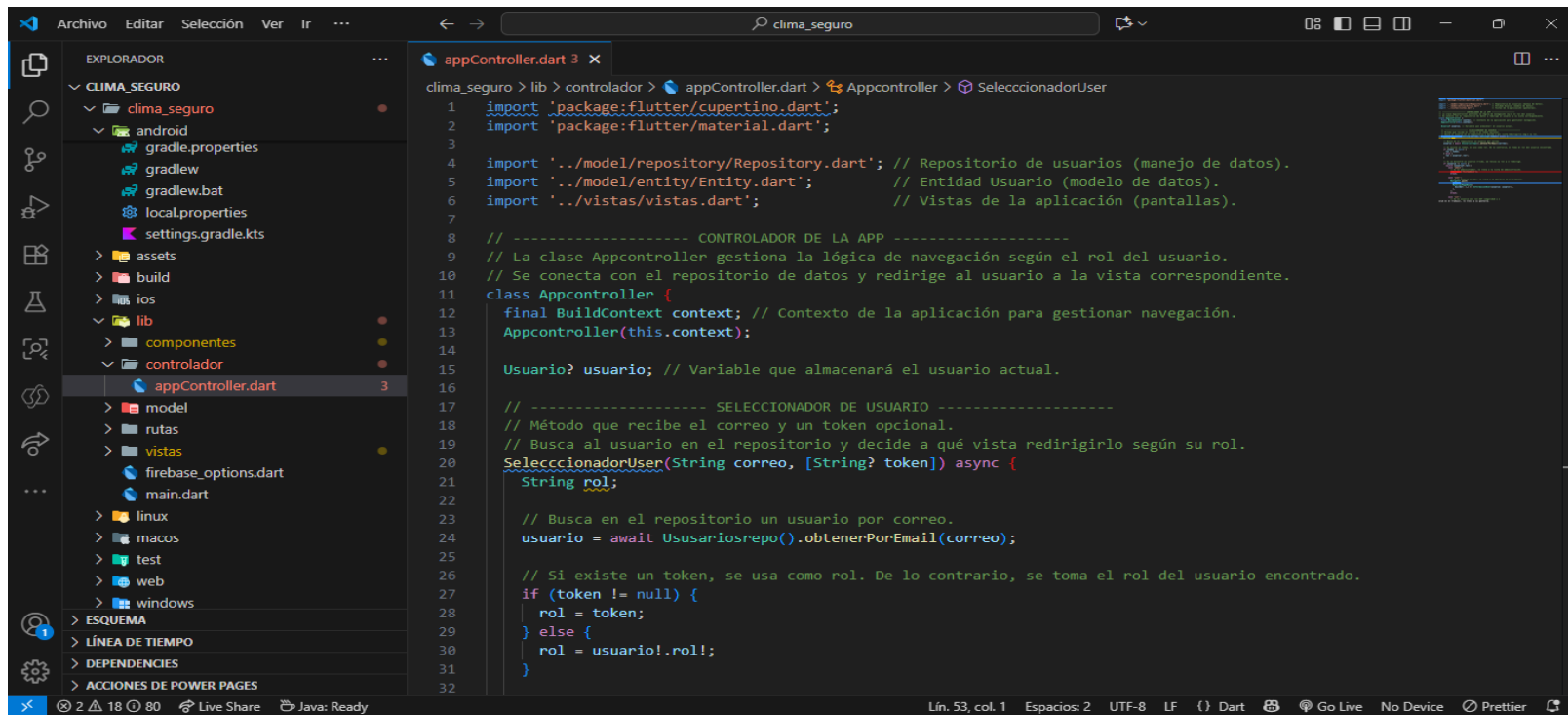
- **Figura 68 Models**

## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

- **Controller**

Este se encarga de obtener y procesar los datos, es decir, traer información de Firebase o de una API de clima.



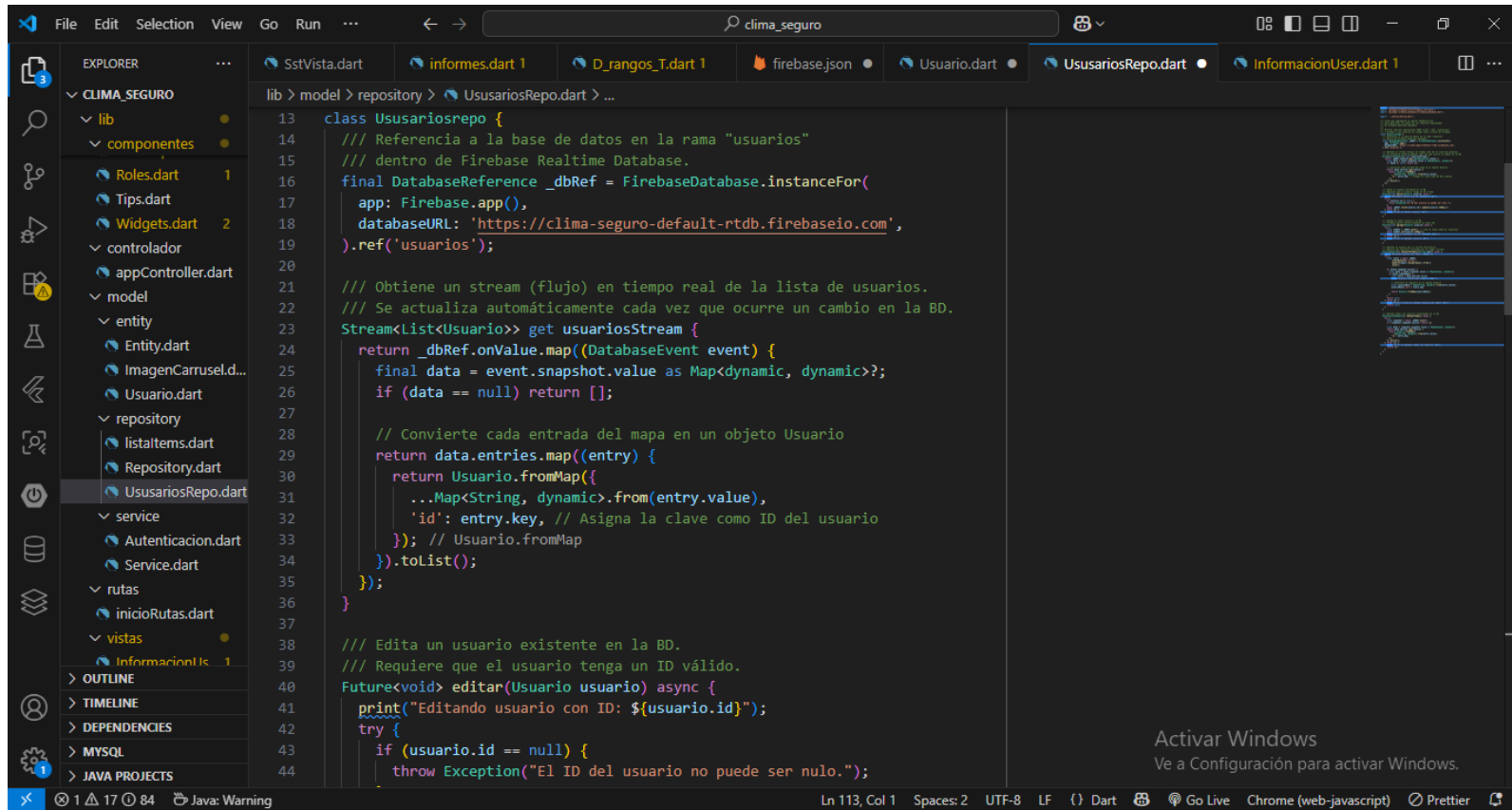
```
clima_seguro > lib > controlador > appController.dart > Appcontroller > SeleccionadorUser
1 import 'package:flutter/cupertino.dart';
2 import 'package:flutter/material.dart';
3
4 import '../model/repository/Repository.dart'; // Repositorio de usuarios (manejo de datos).
5 import '../model/entity/Entity.dart';       // Entidad Usuario (modelo de datos).
6 import '../vistas/vistas.dart';             // Vistas de la aplicación (pantallas).
7
8 // ----- CONTROLADOR DE LA APP -----
9 // La clase Appcontroller gestiona la lógica de navegación según el rol del usuario.
10 // Se conecta con el repositorio de datos y redirige al usuario a la vista correspondiente.
11 class Appcontroller {
12   final BuildContext context; // Contexto de la aplicación para gestionar navegación.
13   Appcontroller(this.context);
14
15   Usuario? usuario; // Variable que almacenará el usuario actual.
16
17 // ----- SELECCIONADOR DE USUARIO -----
18 // Método que recibe el correo y un token opcional.
19 // Busca al usuario en el repositorio y decide a qué vista redirigirlo según su rol.
20 SeleccionadorUser(String correo, [String? token]) async {
21   String rol;
22
23   // Busca en el repositorio un usuario por correo.
24   usuario = await Usuariosrepo().obtenerPorEmail(correo);
25
26   // Si existe un token, se usa como rol. De lo contrario, se toma el rol del usuario encontrado.
27   if (token != null) {
28     rol = token;
29   } else {
30     rol = usuario!.rol!;
31   }
32 }
```

- **Figura 69 Controller**

## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

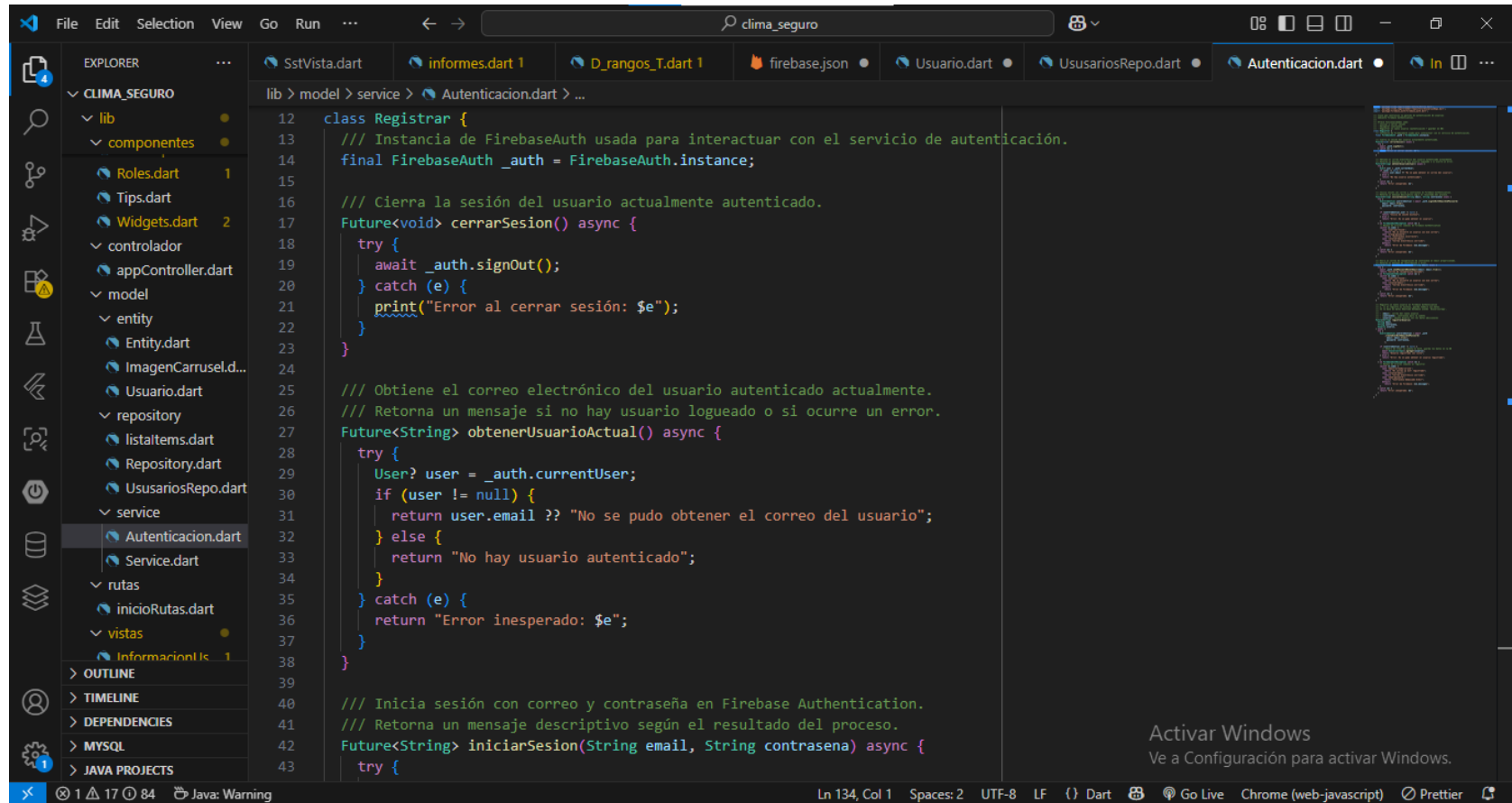
- Model Usuariorepository



```
lib > model > repository > UsuariosRepo.dart > ...
13 class Usuariosrepo {
14   // Referencia a la base de datos en la rama "usuarios"
15   // dentro de Firebase Realtime Database.
16   final DatabaseReference_dbRef = FirebaseDatabase.instanceFor(
17     app: Firebase.app(),
18     databaseURL: 'https://clima-seguro-default-rtdb.firebaseio.com',
19   ).ref('usuarios');
20
21   // Obtiene un stream (flujo) en tiempo real de la lista de usuarios.
22   // Se actualiza automáticamente cada vez que ocurre un cambio en la BD.
23   Stream<List<Usuario>> get usuariosStream {
24     return _dbRef.onValue.map((DatabaseEvent event) {
25       final data = event.snapshot.value as Map<dynamic, dynamic>;
26       if (data == null) return [];
27
28       // Convierte cada entrada del mapa en un objeto Usuario
29       return data.entries.map((entry) {
30         return Usuario.fromMap({
31           ..Map<String, dynamic>.from(entry.value),
32           'id': entry.key, // Asigna la clave como ID del usuario
33         }); // Usuario.fromMap
34       }).toList();
35     });
36   }
37
38   // Edita un usuario existente en la BD.
39   // Requiere que el usuario tenga un ID válido.
40   Future<void> editar(Usuario usuario) async {
41     print("Editando usuario con ID: ${usuario.id}");
42     try {
43       if (usuario.id == null) {
44         throw Exception("El ID del usuario no puede ser nulo.");
45       }
46     } catch (e) {
47       // Manejo de errores
48     }
49   }
50 }
```

- Figura 70 Model Usuariorepository

- Authentication



The screenshot shows an IDE window with the following content:

- EXPLORER:** A file tree for a project named 'CLIMA\_SEGURO'. It includes folders for 'lib', 'componentes', 'controlador', 'model', 'entity', 'repository', 'rutas', and 'vistas'. The 'service' folder is expanded, showing 'Autenticacion.dart' selected.
- Code Editor:** The file 'lib > model > service > Autenticacion.dart' is open. It contains the following Dart code:

```
12 class Registrar {
13     /// Instancia de FirebaseAuth usada para interactuar con el servicio de autenticación.
14     final FirebaseAuth _auth = FirebaseAuth.instance;
15
16     /// Cierra la sesión del usuario actualmente autenticado.
17     Future<void> cerrarSesion() async {
18         try {
19             await _auth.signOut();
20         } catch (e) {
21             print("Error al cerrar sesión: $e");
22         }
23     }
24
25     /// Obtiene el correo electrónico del usuario autenticado actualmente.
26     /// Retorna un mensaje si no hay usuario logueado o si ocurre un error.
27     Future<String> obtenerUsuarioActual() async {
28         try {
29             User? user = _auth.currentUser;
30             if (user != null) {
31                 return user.email ?? "No se pudo obtener el correo del usuario";
32             } else {
33                 return "No hay usuario autenticado";
34             }
35         } catch (e) {
36             return "Error inesperado: $e";
37         }
38     }
39
40     /// Inicia sesión con correo y contraseña en Firebase Authentication.
41     /// Retorna un mensaje descriptivo según el resultado del proceso.
42     Future<String> iniciarSesion(String email, String contrasena) async {
43         try {
```
- Status Bar:** Shows 'Ln 134, Col 1', 'Spaces: 2', 'UTF-8', 'LF', 'Dart', and other IDE settings.
- Bottom Right:** A Windows watermark that says 'Activar Windows. Ve a Configuración para activar Windows.'

- Figura 71 Autenticación

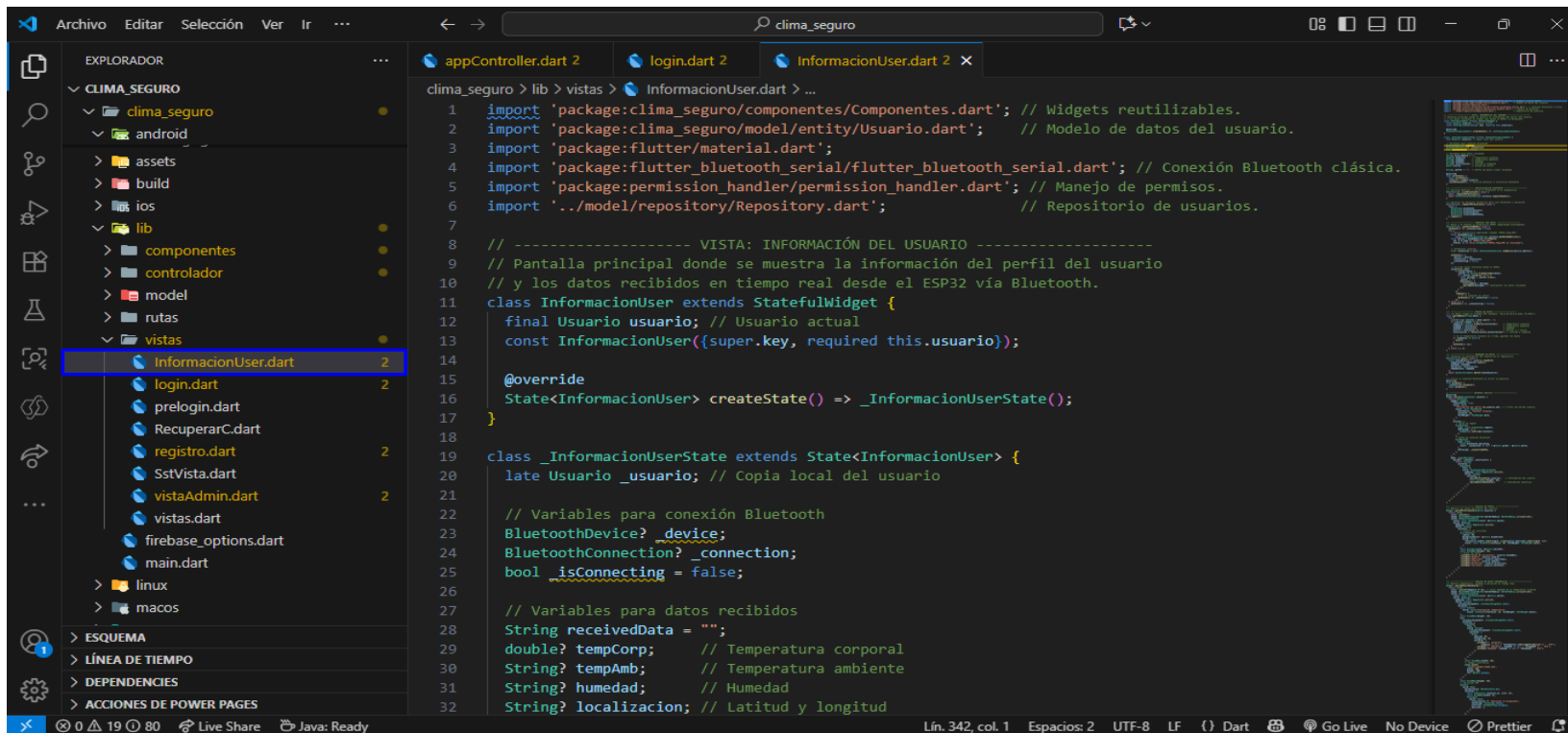
## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

- **Vistas**

Las vistas representan la interfaz gráfica de la aplicación. Aquí se muestran los datos en widgets y se definen las interacciones con el usuario.

La vista principal del usuario logueado, donde se conecta al dispositivo ESP32 por Bluetooth, recibe los datos de los sensores y los muestra junto con la información personal del usuario.



```
clima_seguro > lib > vistas > InformacionUser.dart > ...
1 import 'package:clima_seguro/componentes/Componentes.dart'; // Widgets reutilizables.
2 import 'package:clima_seguro/model/entity/Usuario.dart'; // Modelo de datos del usuario.
3 import 'package:flutter/material.dart';
4 import 'package:flutter_bluetooth_serial/flutter_bluetooth_serial.dart'; // Conexión Bluetooth clásica.
5 import 'package:permission_handler/permission_handler.dart'; // Manejo de permisos.
6 import '../model/repository/Repository.dart'; // Repositorio de usuarios.
7
8 // ----- VISTA: INFORMACIÓN DEL USUARIO -----
9 // Pantalla principal donde se muestra la información del perfil del usuario
10 // y los datos recibidos en tiempo real desde el ESP32 vía Bluetooth.
11 class InformacionUser extends StatefulWidget {
12   final Usuario usuario; // Usuario actual
13   const InformacionUser({super.key, required this.usuario});
14
15   @override
16   State<InformacionUser> createState() => _InformacionUserState();
17 }
18
19 class _InformacionUserState extends State<InformacionUser> {
20   late Usuario _usuario; // Copia local del usuario
21
22   // Variables para conexión Bluetooth
23   BluetoothDevice? _device;
24   BluetoothConnection? _connection;
25   bool _isConnecting = false;
26
27   // Variables para datos recibidos
28   String receivedData = "";
29   double? tempCorp; // Temperatura corporal
30   String? tempAmb; // Temperatura ambiente
31   String? humedad; // Humedad
32   String? localizacion; // Latitud y longitud
```

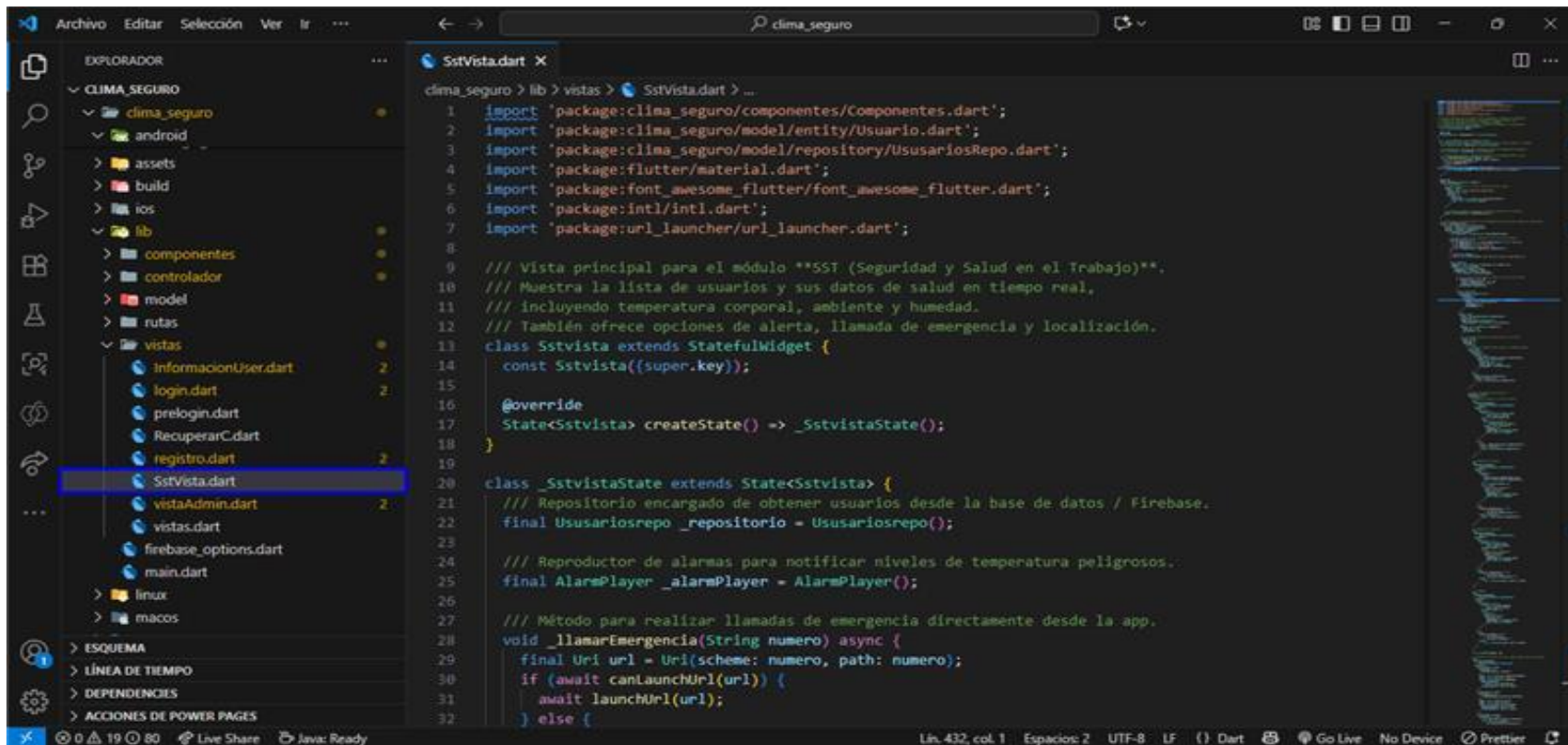
- **Figura 72** vista principal

## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

- Pantalla SST

De la aplicación, que es el módulo de Seguridad y Salud en el Trabajo. Aquí se listan los usuarios en tiempo real con su temperatura corporal, ambiente y humedad.



```
1 import 'package:clima_seguro/componentes/Componentes.dart';
2 import 'package:clima_seguro/model/entity/Usuario.dart';
3 import 'package:clima_seguro/model/repository/UsuariosRepo.dart';
4 import 'package:flutter/material.dart';
5 import 'package:font_awesome_flutter/font_awesome_flutter.dart';
6 import 'package:intl/intl.dart';
7 import 'package:url_launcher/url_launcher.dart';
8
9 /// Vista principal para el módulo **SST (Seguridad y Salud en el Trabajo)**.
10 /// Muestra la lista de usuarios y sus datos de salud en tiempo real,
11 /// incluyendo temperatura corporal, ambiente y humedad.
12 /// También ofrece opciones de alerta, llamada de emergencia y localización.
13 class SstVista extends StatefulWidget {
14   const SstVista({super.key});
15
16   @override
17   State<SstVista> createState() => _SstVistaState();
18 }
19
20 class _SstVistaState extends State<SstVista> {
21   /// Repositorio encargado de obtener usuarios desde la base de datos / Firebase.
22   final Usuariosrepo _repositorio = Usuariosrepo();
23
24   /// Reproductor de alarmas para notificar niveles de temperatura peligrosos.
25   final AlarmPlayer _alarmPlayer = AlarmPlayer();
26
27   /// Método para realizar llamadas de emergencia directamente desde la app.
28   void _llamarEmergencia(String numero) async {
29     final Uri url = Uri(scheme: numero, path: numero);
30     if (await canLaunchUrl(url)) {
31       await launchUrl(url);
32     } else {
```

- Figura 73 Pantalla SST

- **Vista Admin**

Es una interfaz para el usuario administrador, que le permite gestionar diferentes módulos del sistema como roles, dispositivos, tips, historial y empleados. Es una vista central de control para supervisar la información registrada en la aplicación.

```

1  import 'package:clima_seguro/componentes/listaEmpleados.dart';
2  import 'package:flutter/material.dart';
3  import 'package:font_awesome_flutter/font_awesome_flutter.dart';
4  import '../componentes/Componentes.dart';
5  import '../componentes/Tips.dart';
6  import '../model/entity/Usuario.dart';
7  import '../model/repository/Repository.dart';
8
9  /// -----
10 /// Vista administrativa principal
11 /// -----
12 /// Esta pantalla es el panel de control para el administrador.
13 /// Desde aquí se seleccionan módulos: Roles, Dispositivos, Tips,
14 /// Historial y Empleados. El contenido cambia dinámicamente según
15 /// la opción seleccionada.
16 /// -----
17 class Vistaadmin extends StatefulWidget {
18   const Vistaadmin({super.key});
19
20   @override
21   State<Vistaadmin> createState() => _VistaadminState();
22 }
23
24 class _VistaadminState extends State<Vistaadmin> {
25   // Variable que guarda la opción actualmente seleccionada en el menú superior.
26   // Por defecto mostramos 'Roles'.
27   String _selectedOption = 'Roles';
28
29   @override
30   Widget build(BuildContext context) {
31     return Scaffold(
32       backgroundColor: Colors.white, // Fondo blanco para enfoque en contenido.

```

- **Figura 74 Vista Admin**

- **Vista Inicio de sesión**

En esta vista se implementa el formulario de inicio de sesión en la aplicación. Permite al usuario ingresar su correo y contraseña, valida los datos, y luego usa el servicio Registrar().iniciarSesion para autentificarlo. Dependiendo de su rol (usuario, administrador o SST), el controlador (Appcontroller) lo redirige a la vista correspondiente. Además, muestra mensajes de error o éxito con Snackbar y ofrece accesos rápidos al registro de cuenta y a la recuperación de contraseña.

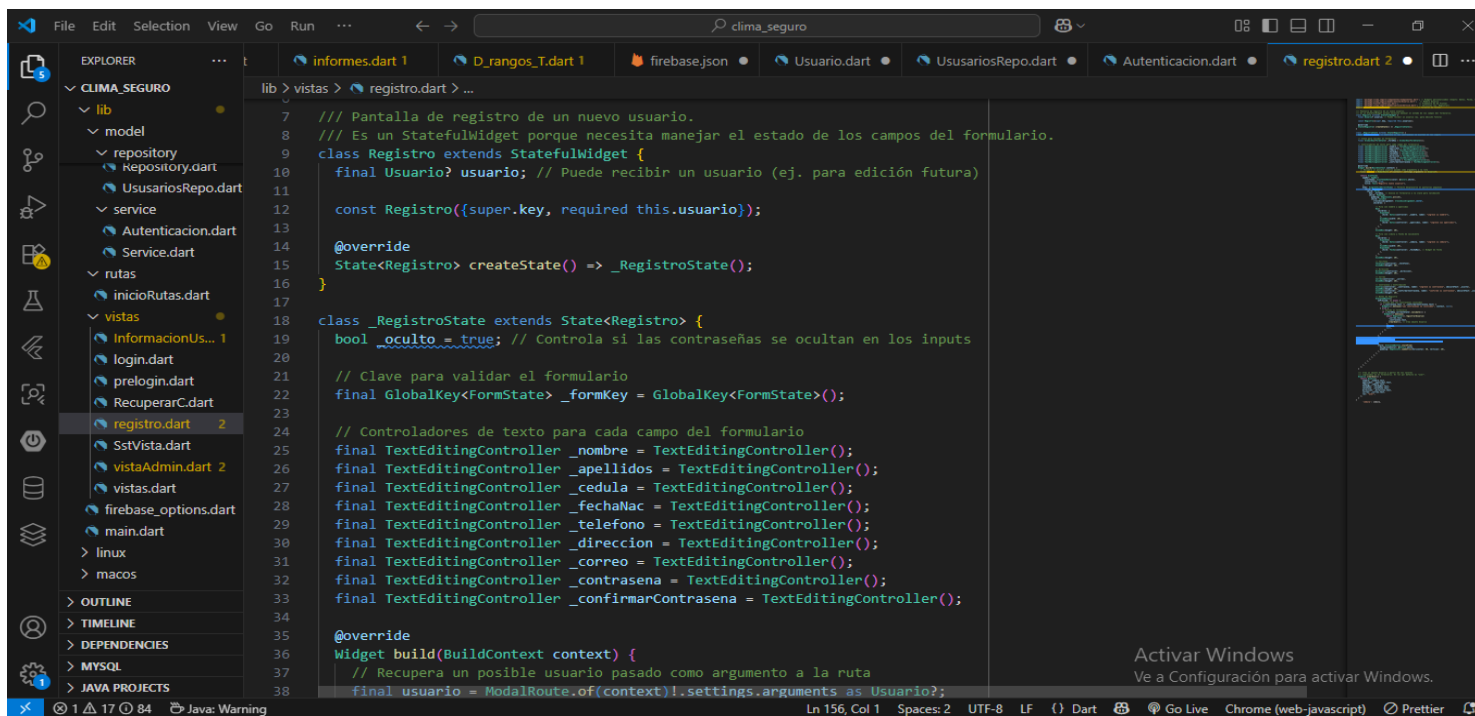
The screenshot shows an IDE window with the following content:

- EXPLORADOR (Left Panel):** A file explorer showing the project structure. The 'vistas' folder is expanded, and 'login.dart' is selected.
- Code Editor (Center):** The Dart code for 'login.dart'. It includes imports for 'dart:io', 'package:clima\_seguro/controlador/appController.dart', 'package:clima\_seguro/model/entity/Usuario.dart', 'package:clima\_seguro/model/repository/UsuariosRepo.dart', 'package:flutter/material.dart', 'package:clima\_seguro/model/service/Service.dart', and 'package:clima\_seguro/vistas/InformacionUser.dart'. The code defines a 'Login' class extending 'StatefulWidget' and a '\_LoginState' class extending 'State<Login>'. Comments describe the view's purpose and state management.
- Inspector (Right Panel):** A widget inspector showing the widget tree for the login view.
- Status Bar (Bottom):** Shows 'Lín. 7, col. 59', 'Espacios: 2', 'UTF-8', 'LF', 'Dart', 'Go Live', 'No Device', and 'Prettier'.

- **Figura 75 Inicio de sesión**

- Registro

En este código se implementa la pantalla de registro del usuario, la cual contiene un formulario que recopila la información necesaria para hacer el registro (nombre, apellidos, cédula, fecha de nacimiento, teléfono, dirección, correo y contraseña), se hace una validación de los datos ingresados y al confirmarse se crea un objeto Usuario con esa información y lo envía al servicio Registrar, que lo almacena en Firebase Authentication y en la base de datos en tiempo real.



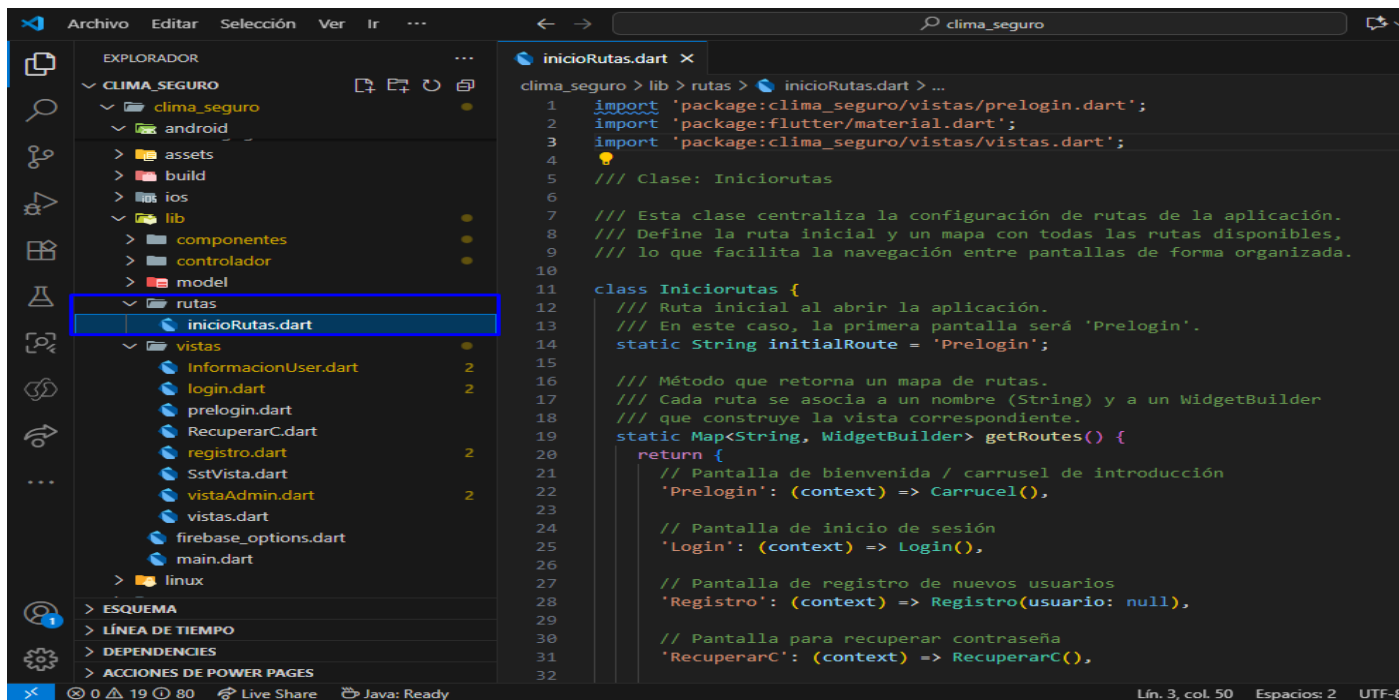
```
lib > vistas > registro.dart > ...
7 // Pantalla de registro de un nuevo usuario.
8 // Es un StatefulWidget porque necesita manejar el estado de los campos del formulario.
9 class Registro extends StatefulWidget {
10   final Usuario? usuario; // Puede recibir un usuario (ej. para edición futura)
11
12   const Registro({super.key, required this.usuario});
13
14   @override
15   State<Registro> createState() => _RegistroState();
16 }
17
18 class _RegistroState extends State<Registro> {
19   bool _oculto = true; // Controla si las contraseñas se ocultan en los inputs
20
21   // Clave para validar el formulario
22   final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
23
24   // Controladores de texto para cada campo del formulario
25   final TextEditingController _nombre = TextEditingController();
26   final TextEditingController _apellidos = TextEditingController();
27   final TextEditingController _cedula = TextEditingController();
28   final TextEditingController _fechaNac = TextEditingController();
29   final TextEditingController _telefono = TextEditingController();
30   final TextEditingController _direccion = TextEditingController();
31   final TextEditingController _correo = TextEditingController();
32   final TextEditingController _contrasena = TextEditingController();
33   final TextEditingController _confirmarContrasena = TextEditingController();
34
35   @override
36   Widget build(BuildContext context) {
37     // Recupera un posible usuario pasado como argumento a la ruta
38     final usuario = ModalRoute.of(context)?.settings.arguments as Usuario?;
```

- Figura 76 Registro

## Rutas

Aquí se centraliza la navegación entre pantallas de la aplicación.

La clase Inicio rutas funciona como el mapa de navegación de la aplicación, ya que define la pantalla inicial (Prelogin) y centraliza todas las rutas que permiten moverse entre las diferentes vistas, como el inicio de sesión, el registro, la recuperación de contraseña, el panel de administrador y la vista de SST. Gracias a este archivo, la app mantiene organizado el flujo de navegación y facilita la gestión de las pantallas, ya que cada ruta tiene un nombre único asociado a su respectiva vista.



```

1  import 'package:clima_seguro/vistas/prelogin.dart';
2  import 'package:flutter/material.dart';
3  import 'package:clima_seguro/vistas/vistas.dart';
4
5  /// Clase: Iniciorutas
6
7  /// Esta clase centraliza la configuración de rutas de la aplicación.
8  /// Define la ruta inicial y un mapa con todas las rutas disponibles,
9  /// lo que facilita la navegación entre pantallas de forma organizada.
10
11 class Iniciorutas {
12   /// Ruta inicial al abrir la aplicación.
13   /// En este caso, la primera pantalla será 'Prelogin'.
14   static String initialRoute = 'Prelogin';
15
16   /// Método que retorna un mapa de rutas.
17   /// Cada ruta se asocia a un nombre (String) y a un WidgetBuilder
18   /// que construye la vista correspondiente.
19   static Map<String, WidgetBuilder> getRoutes() {
20     return {
21       // Pantalla de bienvenida / carrusel de introducción
22       'Prelogin': (context) => Carrusel(),
23
24       // Pantalla de inicio de sesión
25       'Login': (context) => Login(),
26
27       // Pantalla de registro de nuevos usuarios
28       'Registro': (context) => Registro(usuario: null),
29
30       // Pantalla para recuperar contraseña
31       'RecuperarC': (context) => RecuperarC(),
32     };
33   }
34 }

```

- **Figura 77 Rutas**

## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

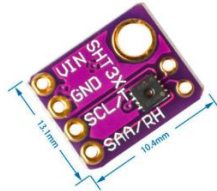
#### 26.4 Componentes

##### Sensores y Electrónica

- **Sensor de temperatura y humedad (SHT31):**

Este sensor permite realizar mediciones muy precisas a bajo costo. Le ofrece una mejor calidad que los sensores DHT11 y DHT22, especialmente mejor resolución, mayor precisión y compactación de empaque. Se emplea en aplicaciones usadas en temperatura, tales como control automático, aire acondicionado, monitoreo en agricultura, entre otros.

Sencillez de su integración, tanto en componentes software como hardware. En este sentido, en lo que respecta a software, se dispone de librerías para Arduino con soporte para el protocolo I2C. En lo que respecta a hardware, para su uso solo son necesarios 4 cables: dos de alimentación y dos de datos I2C.  
(Macatronica Electronica y tecnologia desde 2014, s.f.)



##### ESPECIFICACIONES TECNICAS:

- Voltaje de Operación: 2.4V a 5.5V DC
- Interfaz de comunicación: I2C
- Dirección I2C: 0x44
- Rango de trabajo Temperatura: -40° a 125°C
- Resolución Temperatura: 0.015°C
- Precisión Temperatura: 0.2°C
- Rango de trabajo Humedad: 0 a 100% RH
- Resolución HR: 0.01 %RH
- Precisión HR: 2% RH
- Tiempos de muestreo rápidos

- **Sensor de temperatura corporal (MLX90614):**

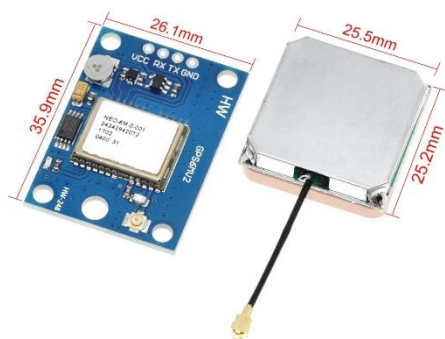
Es un sensor de temperatura infrarrojo sin contacto, ideal para medir la temperatura de objetos a distancia. Utiliza una termopila MEMS que detecta la radiación infrarroja y convierte la señal en una lectura digital. El sensor incluye un filtro de ruido, conversor A/D de 17 bits y un DSP, lo que mejora la precisión. Para utilizarlo con Arduino o ESP32, es necesario identificar su dirección I2C, lo que puede hacerse con un I2C Scanner. En comparación con el MLX90614ESF-BAA, este modelo tiene mayor precisión y mejor compensación térmica, lo que lo hace más fiable en entornos variables. (Electronica plug and play, s.f.)



#### ESPECIFICACIONES TECNICAS

- Voltaje de operación: 3.3V-5V DC
- Protocolo de comunicación: SMBUS (subconjunto del I2C)
- Rango de temperatura ambiental: -40°C hasta +85°C
- Rango de temperatura de objeto: -70°C hasta +380°C
- Precisión:  $\pm 0.2^{\circ}\text{C}$
- Ángulo de visión: 5° (FOV)
- ADC interno de 17 bits
- Procesador digital de señal interno (DSP)
- Regulador de voltaje 3.3V en placa
- Resistencias Pull-up a VIN en placa
- No necesita componentes adicionales
- Peso: 5 gramos

- **Módulo GPS (NEO-6M):** Diseñado con base al módulo U-Blox 6 un motor de posicionamiento de 50 canales cuenta con un tiempo hasta la primera reparación (TTFF) de menos de 1 segundo, ofreciendo gran rendimiento y estabilidad. Capaz de realizar búsquedas masivas de espacio de tiempo / frecuencia en paralelo, adquisición, con hasta 2 millones de adquisiciones correlacionadas permitiendo encontrar satélites al instante. Incluye una memoria EEPROM para poder almacenar datos de configuración del módulo. (Sigma electronica, s.f.)



#### ESPECIFICACIONES TECNICAS

- Voltaje de alimentación: 3.3 V ~ 5.0 V.
- Consumo de corriente: ~ 100 mA.
- Recepción: 50 Canales; GPS L1 frequency, C/A Code, SBAS: WAAS, EGNOS, MSAS.
- Tiempo de enganche promedio: 1 s ~ 27 s.
- Precisión de Posicionamiento: 2.5 MCEP.
- Protocolo de comunicación: NMEA, UBX binario.
- Velocidad de comunicación: 9600 baudios.

- **Batería recargable LiPo (3.7V, 2500mAh):** Bateria Recargable 18650 3.7V - 2500 mAh, ampliamente utilizada en juguetes, encendedores, equipos de comunicación, linternas, power bank y diversos dispositivos electrónicos. (Pilas RECARGABLES 3.7 V 2.5 Ah Amp horas, s.f.)



#### ESPECIFICACIONES TECNICAS

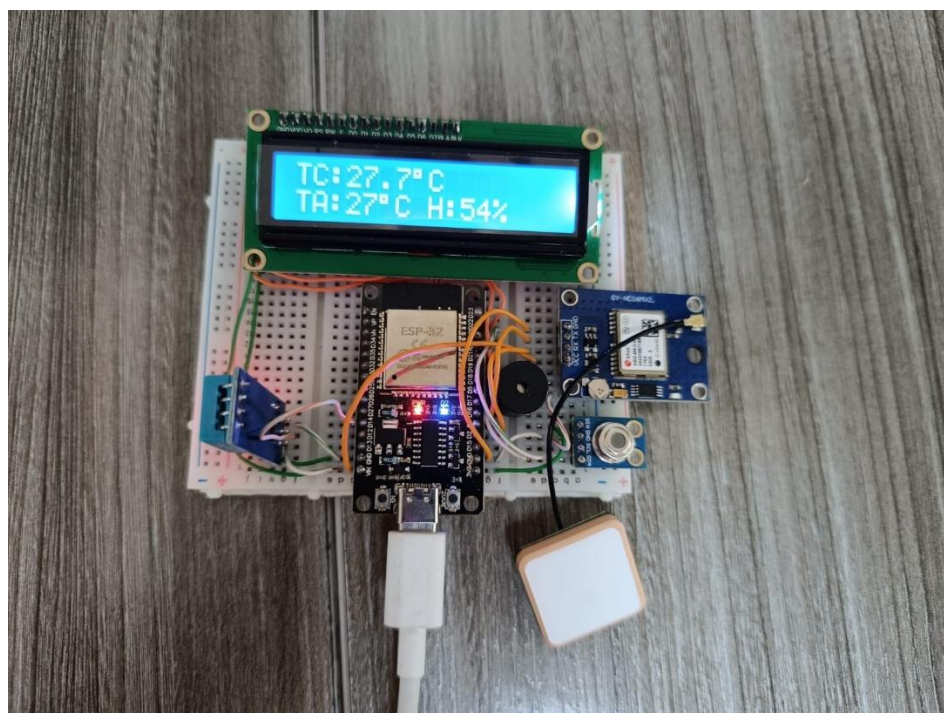
- Marca: Hi-tronic
- Modelo de la Bateria (Pila): 18650
- Voltaje: 3.7V
- Corriente: 2500mAh
- Material: Ion de Litio
- Peso: ~ 45 gr
- Medidas: 65 mm x 18 mm

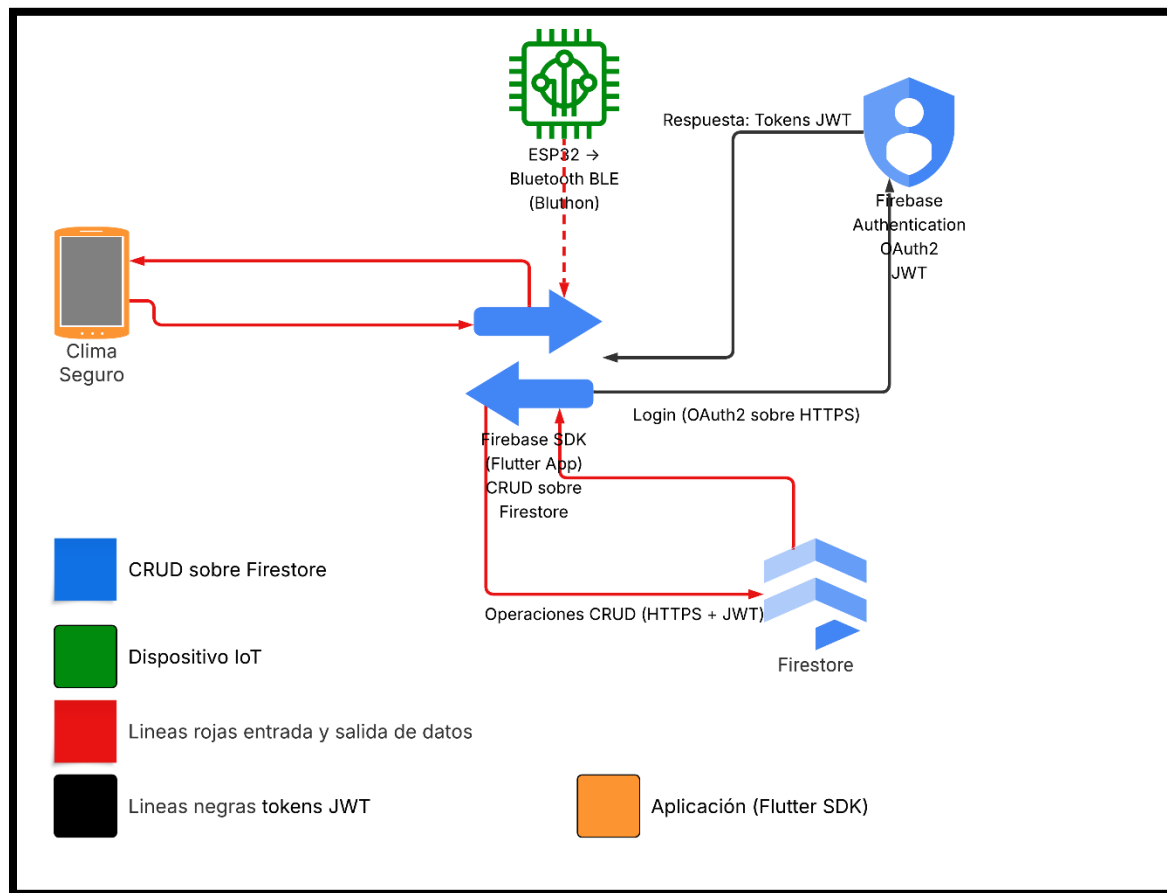
**Microcontrolador (ESP32):** es un potente módulo que integra WiFi y Bluetooth, ideal para desarrollar productos de IoT. La integración de Bluetooth, Bluetooth LE y Wi-Fi permite una amplia gama de aplicaciones, el uso de WiFi permite una comunicación de mediano alcance y conectarse a una red LAN y a través de un Modem Router conexión a Internet, mientras que el Bluetooth nos permite conectarse directamente a otro dispositivo como un celular. (Sigma Electronica, s.f.)



#### ESPECIFICACIONES TECNICAS

- Microcontrolador: Xtensa LX6 de 32 bits, dual-core (o single-core)
- Tensión de funcionamiento: 3.3V (máx. 3.6V)
- Pines de E/S digitales: Hasta 34
- Pines de E/S digitales con PWM: Hasta 16 canales
- Pines de entrada analógica: 15 (12-bit SAR ADC)
- SPI/I2C/I2S/UART: 4/2/2/3
- Memoria flash: Hasta 16 MB (externa), 448 KB (interna)
- SRAM: 520 KB
- Velocidad de reloj: 160 MHz o 240 MHz
- Wi-Fi: Sí, 802.11 b/g/n
- Bluetooth: Sí, v4.2 BR/EDR y BLE
- Sensor táctil: 10 sensores capacitivos





- **Figura 78 Integración Hardware**

## 26.5 Integración de Componentes

En las siguientes imágenes se presenta la integración de los componentes con sus respectivos estados de temperatura, adicional se muestra cómo llega la alerta al personal de SST para su respectivo proceso con el usuario.

En esta imagen observaremos la temperatura de un usuario en estado normal.



- **Figura 79** Medición de temperatura

En la imagen se observa la alerta naranja de un usuario dado que su temperatura alcanza los 37°C.



- **Figura 80** Alerta naranja

En la imagen tenemos un usuario con alerta crítica, su temperatura superó los 39°C, en este caso se envía una alerta al personal de SST para su respectiva atención.



- **Figura 81** Alerta Roja

- Notificación de alertas a personal SST



- **Figura 82** Notificación a SST

## 26.6 Evidencia de Procesos Principales

En las siguientes capturas encontremos el flujo completo, Inicio de sesión hacia la aplicación.

### Proceso Registro de Usuario

#### 1. Inicio de Sesión



The screenshot shows a login interface with a dark blue background. At the top is a circular logo with a yellow hard hat and the text 'CLIMA SEGURO' above a cityscape. Below the logo are two input fields: 'ingrese su correo electronico' and 'ingrese su contraseña' with an eye icon. A white 'Iniciar sesion' button is centered below. At the bottom, there are links: 'No tienes cuenta? [Registrate](#)' and 'Olvidaste tu contraseña? [Recuperar contraseña](#)'.

- **Figura 84 Inicio de sesión**

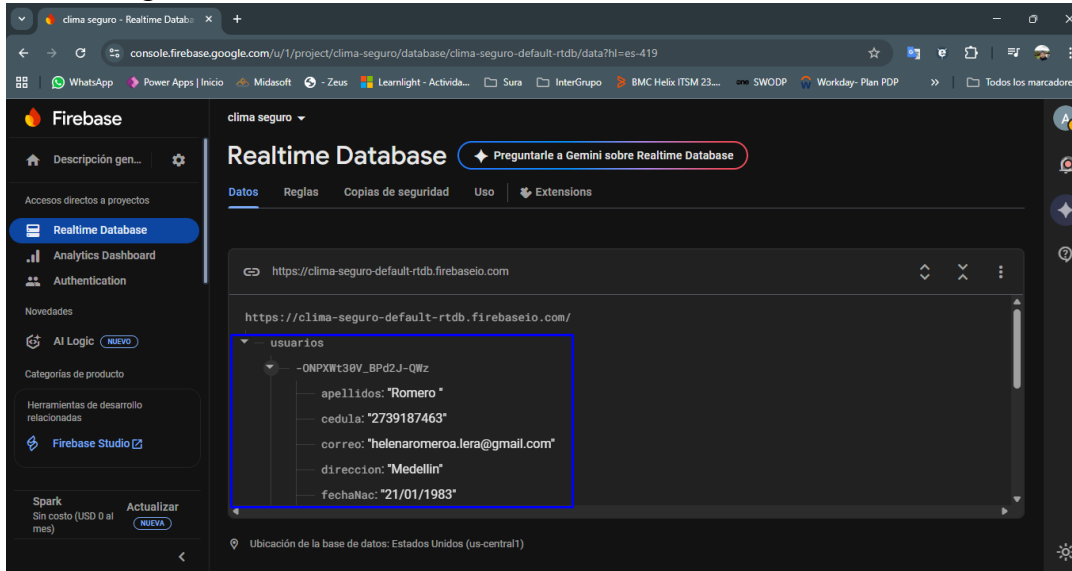
#### 2. Registro de Usuario



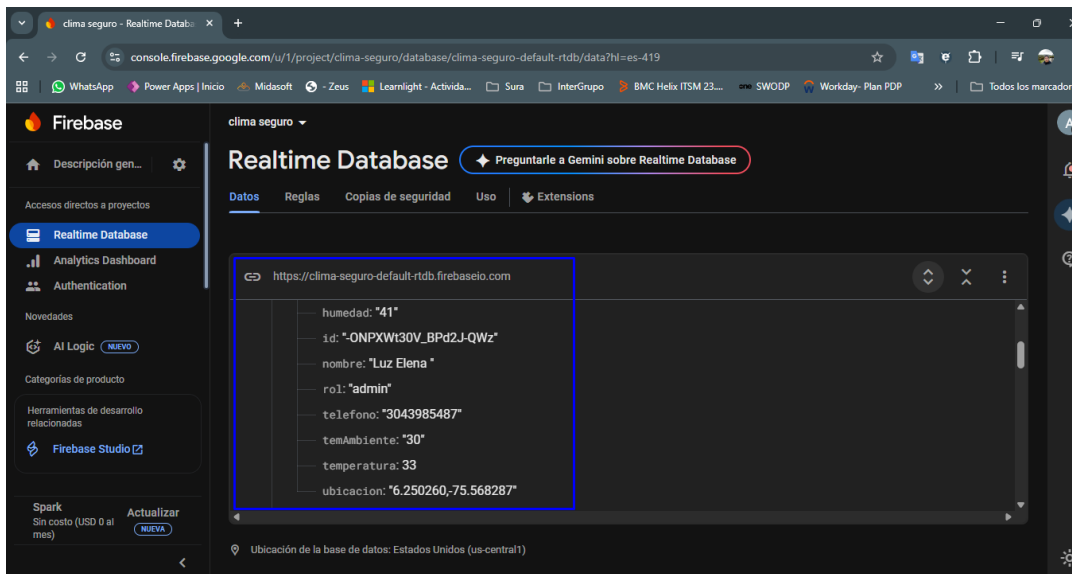
The screenshot shows a registration form titled 'Registro nuevo usuario' on a dark blue background. It includes several input fields: 'ingrese su nombre', 'ingrese sus apellidos', 'ingrese su cedula', 'Fecha de na...' with a calendar icon, 'ingrese telefono', 'ingrese su direccion', 'ingrese su correo electronico', 'ingrese su contraseña' with an eye icon, and 'confirme su contraseña' with an eye icon. A white 'Registrarse' button is at the bottom. The top status bar shows '10:54 AM | 13.7kB/s' and '89%' battery.

- **Figura 83 Registro Usuario**

## Usuario registrado en Base de Datos Firebase



• **Figura 85** Base de Datos Firebase

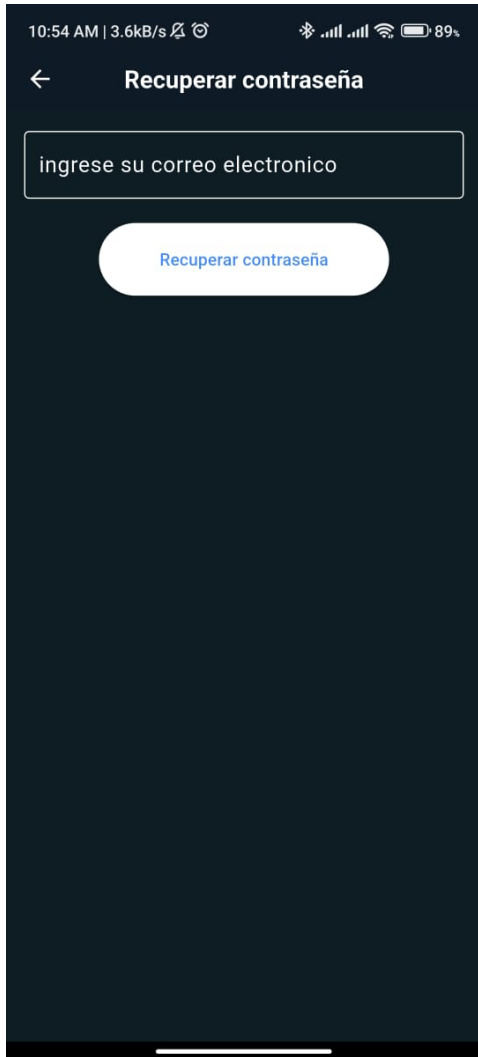


• **Figura 86** Base de Datos Firebase

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**

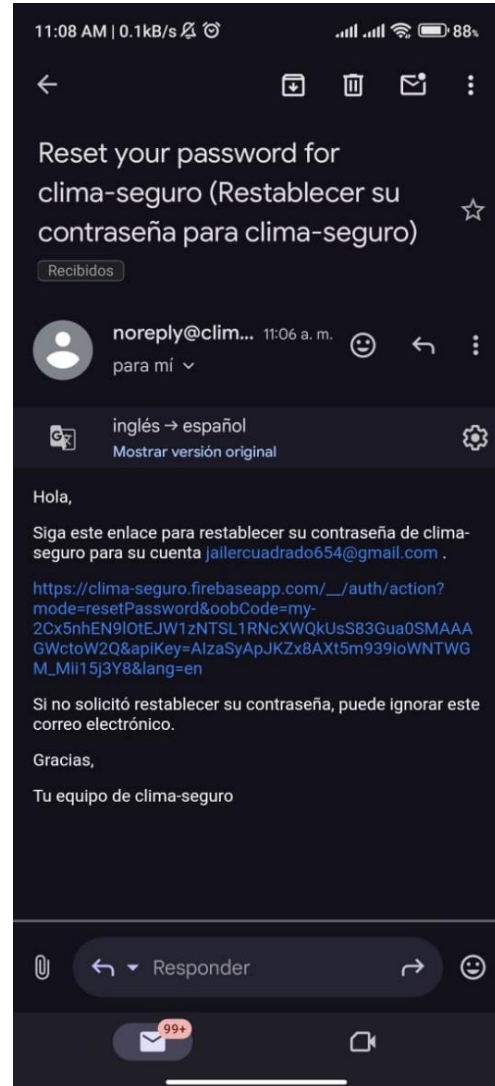
- **Proceso Cambio de contraseña**

1. Recuperar Contraseña



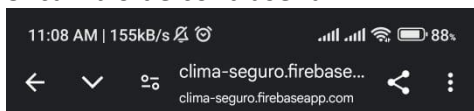
• **Figura 87** Recuperar contraseña

2. Link Recuperar



• **Figura 88** Link recuperar

### 3. cambio de contraseña



#### Contraseña cambiada

Ya puedes iniciar sesión con tu nueva contraseña

### 4. Ingreso nueva contraseña



#### Reset your password

for jailercuadrado654@gmail.com

New password

SAVE

- **Figura 89** Cambio de contraseña

- **Figura 90** ingreso

## Proceso inicio de sesión de Usuario

### 1. Vista principal



• **Figura 91** Vista Principal

### 2. Inicio de Sesión



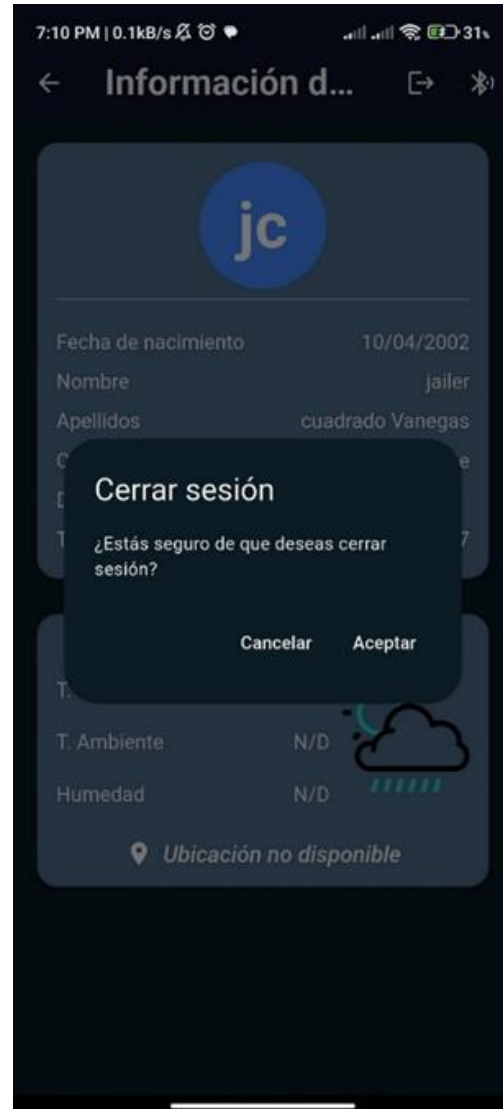
• **Figura 92** Inicio de Sesión

### 3. Perfil del usuario



- **Figura 93** Perfil Usuario

### 4. Cierre de Sesión



- **Figura 94** Cierre de Sesión

## Proceso perfil SST

### 1. Inicio perfil SST



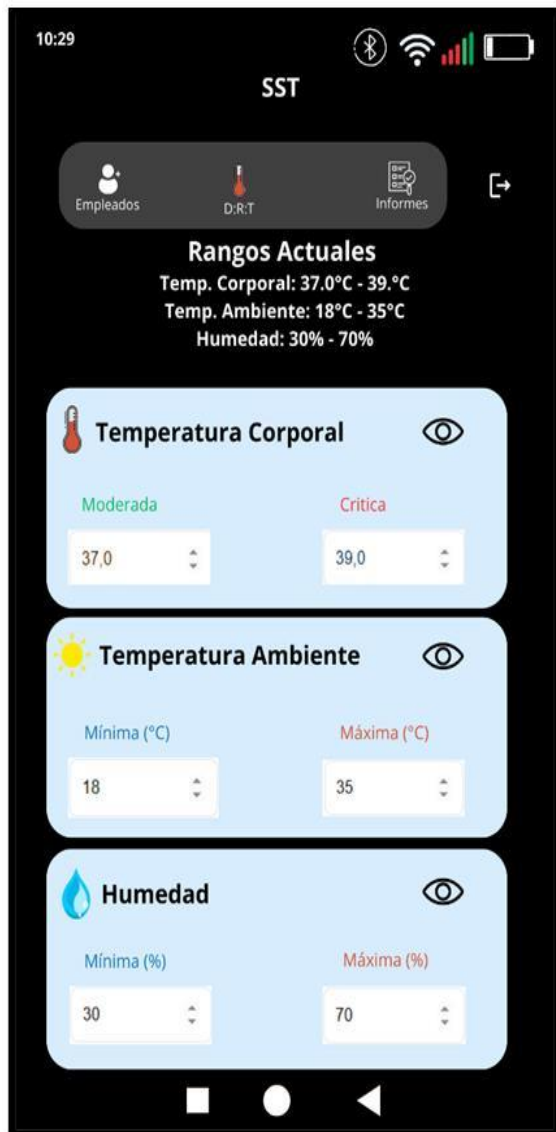
- **Figura 95** Perfil SST

### 2. Vista monitoreo



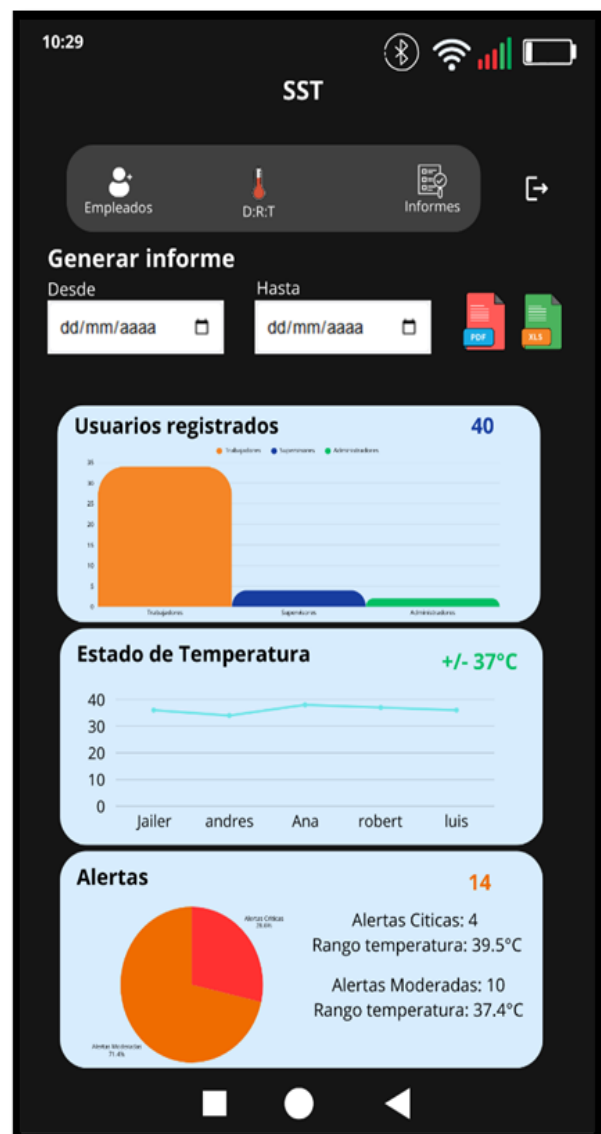
- **Figura 96** Monitoreo

## 4. Pantalla temperatura



- **Figura 97** Temperatura

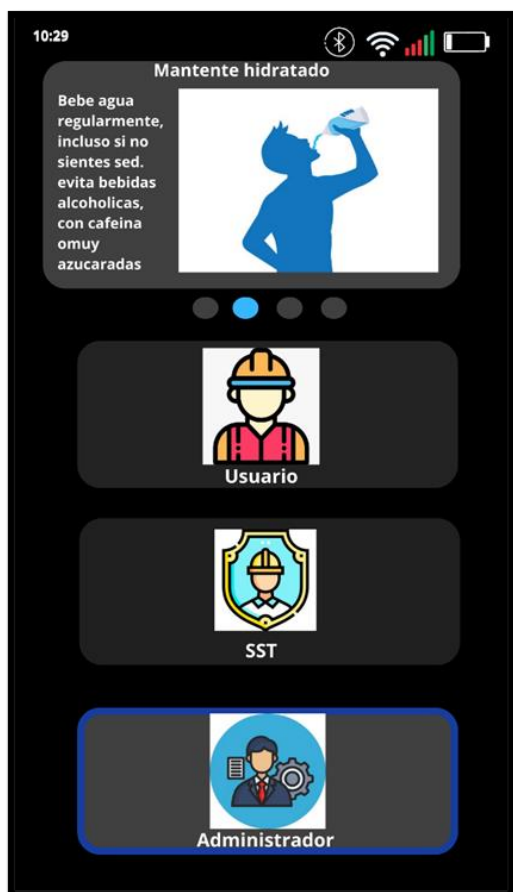
## 5. Pantalla de Informes



- **Figura 98** Informes y consultas

## Proceso perfil administrador

### 1. Inicio perfil administrador



- **Figura 99** Perfil Administrador

### 2. Gestion de Roles



- **Figura 100** Roles

### 3. Gestión de Tips



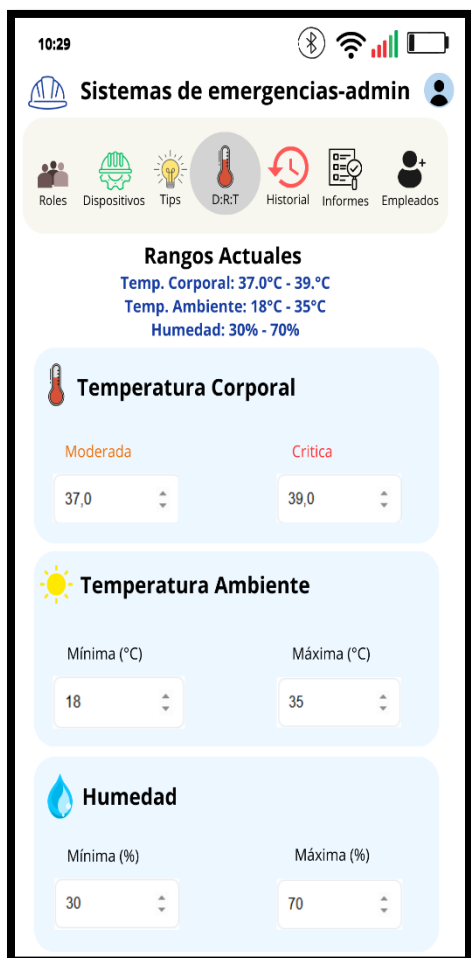
- Figura 101 Tips

### 4. Asignación Dispositivo



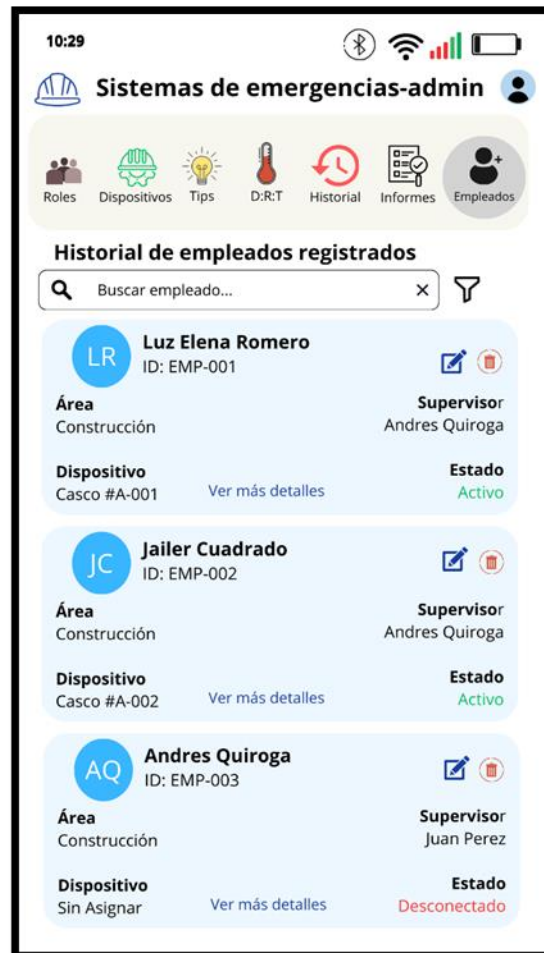
- Figura 102 Asignación Dispositivo

## 5. Rangos de temperatura



- Figura 103 Rangos temperatura

## 6. Historial Golpes de calor

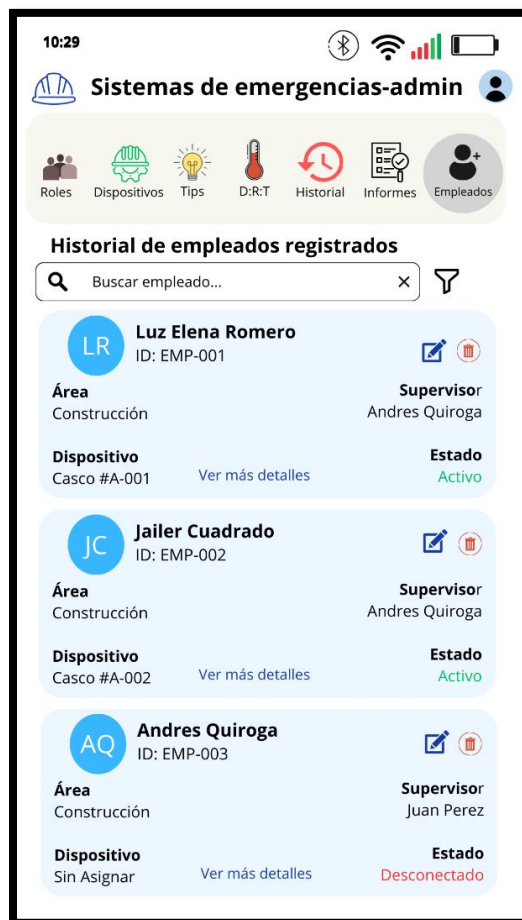


- Figura 104 Historiales golpes de calor

### 7. Informes y consultas



### 8. Historial empleados registrados



## 26.7 Documentación Técnica

Pantallazos de la documentación generada, manual de usuario, manual técnico.

### 27. Manual de Usuario

#### 1. Propósito y alcance del manual

El siguiente manual tiene como objetivo guiar a los usuarios o trabajadores en el uso de la aplicación para medir su temperatura y evitar futuros inconvenientes de salud, por las condiciones de oleadas de calor fuertes.

El casco inteligente le permitirá tomar la temperatura ambiente y enviar alertas en tiempo real a los encargados de SST, adicional enviará su ubicación para que sea más fácil llegar donde está el trabajador.

El manual es principalmente para tres tipos de usuarios:

Trabajadores que son los que usan el casco.

personal de Seguridad y Salud en el trabajo (SST) que estarán pendientes de alertas

Administrador del Sistema: es el que da las instrucciones para que la operación del dispositivo envíe la información en tiempo real al personal encargado.

#### 2. Usuarios y roles

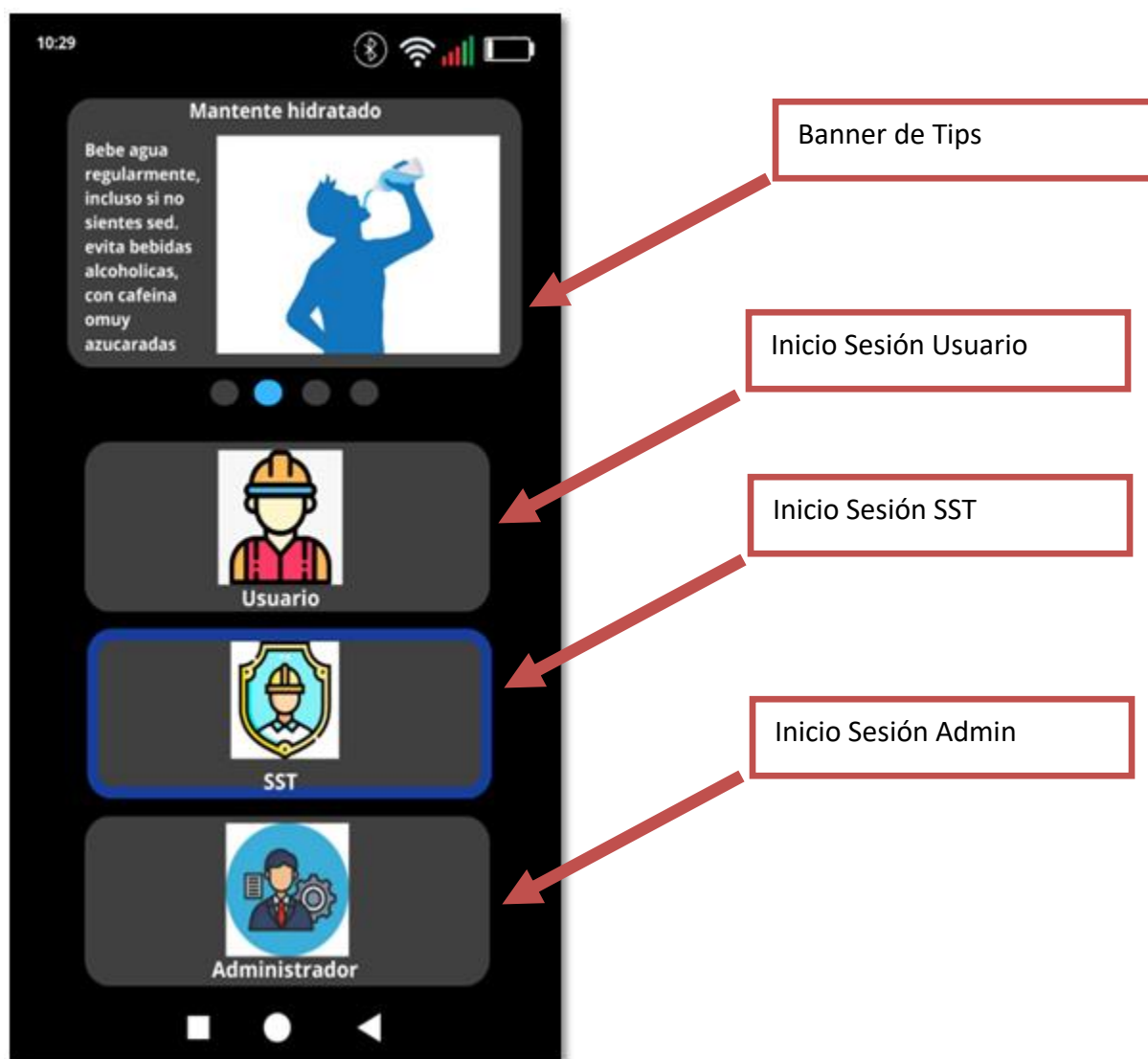
- Trabajador: esta persona debe de portar todo el tiempo el casco puesto en su cabeza para que los datos que arroje la aplicación sean reales y de esta manera tener un buen control por parte de Seguridad y salud en el trabajo.
- Seguridad y Salud en el Trabajo (SST): es la persona encargada de monitorear los datos que se recopilan en la aplicación por medio del casco inteligente que usa cada trabajador, ellos pueden tener el acceso a sacar historiales de las alertas por trabajador, puede tomar decisiones y coordinar acciones de prevención o de emergencias.
- Administrador del Sistema: Él es el encargado de gestionar la plataforma, administrar los perfiles, controlar los accesos, configurar niveles de temperatura para que la aplicación envíe las alertas al personal encargado, el

se encarga de supervisar los reportes de alertas y actividades del sistema, esto con el fin de que la operación envíe información acertada, adicional debe de estar pendiente de la seguridad del sistema.

- Cada uno de estos roles tiene acceso y permisos restringidos dentro de la aplicación o el sistema

## 1 paso a paso para el ingreso de la aplicación

- Vista Principal



- Registro de Usuario-Trabajador



Ingresar perfil Usuario



Seleccionar Registrarse

10:54 AM | 13.7kB/s | 89%

### Registro nuevo usuario

ingrese su nombre

ingrese sus apellidos

ingrese su cedula

Fecha de na...

ingrese telefono

ingrese su direccion

ingrese su correo electronico

ingrese su contraseña

confirme su contraseña

**Registrarse**

Es importante diligenciar toda la información solicitada



Seleccionar Registrarse





Después de hacer el registro, el trabajador ingresa con correo y contraseña

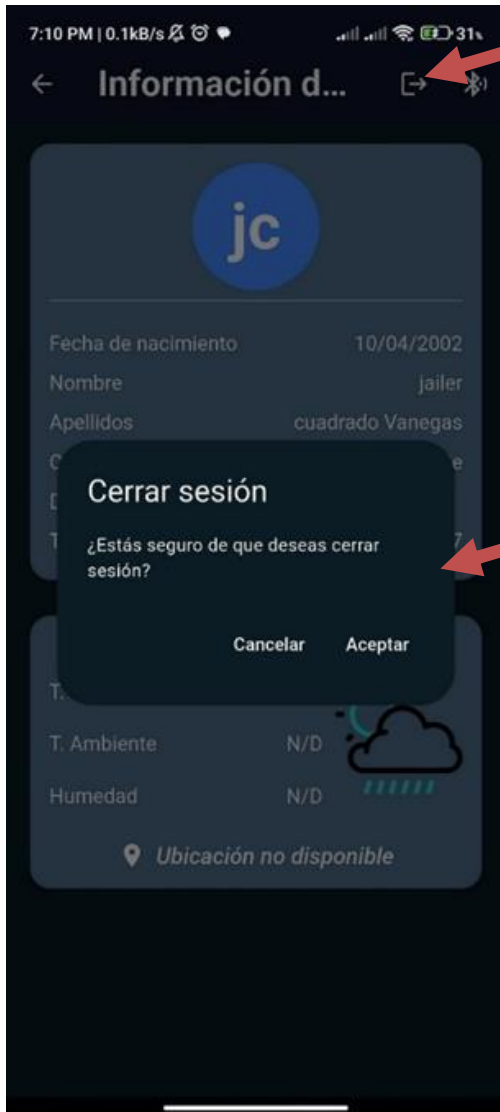
Iniciar Sesión



Una vez inicie Sesión, el trabajador deberá conectar el dispositivo(casco) vía bluetooth para empezar a capturar sus datos de

En esta parte aparecerá la información del trabajador

Una vez se establece la conexión con el dispositivo ya no vera N/D, si no los valores correspondientes a la temperatura y su respectiva ubicación.



Para cerrar la sesión se debe dar clic en esta opción.

Luego se desplegará este mensaje para confirmar el cierre de sesión, solo debe dar clic en ACEPTAR y de esta forma su sesión será cerrada.

**EL CIERRE DE SESION ES IGUAL EN TODOS LOS PERFILES.**

En caso de Olvidar la contraseña se realizarán los siguientes pasos:



ingrese su correo electronico

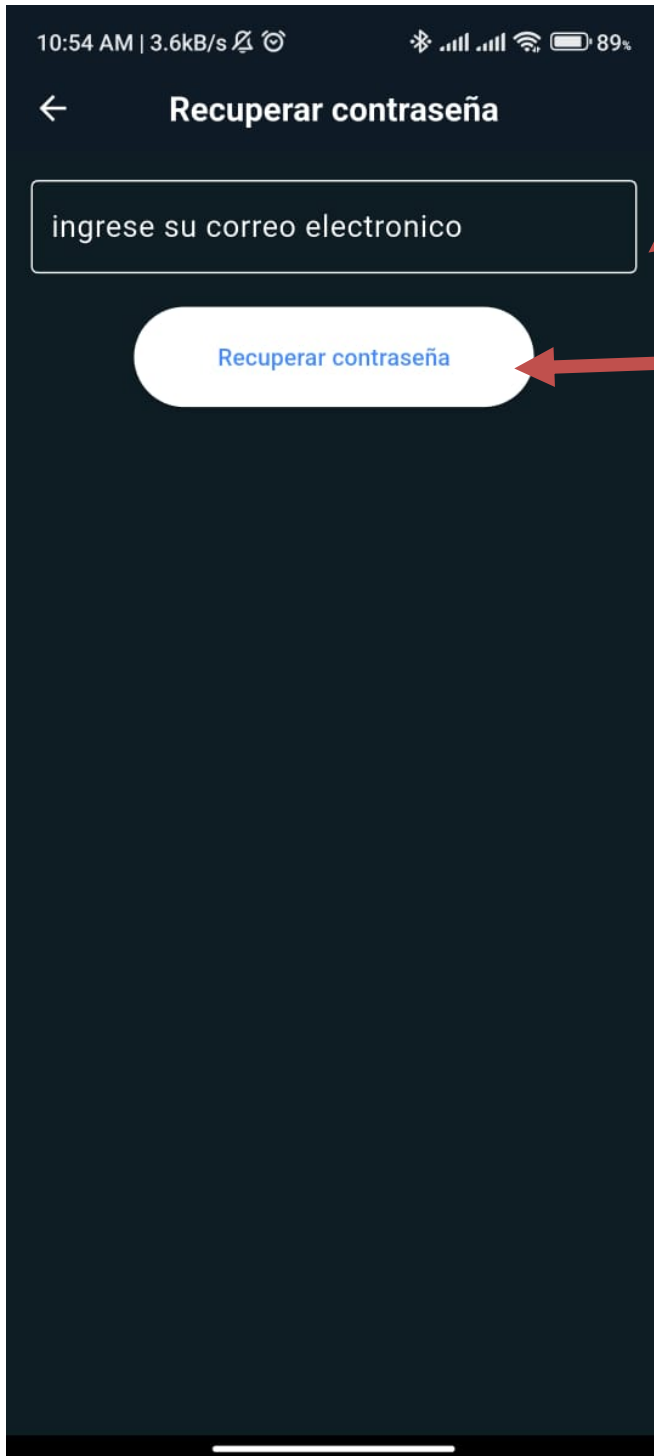
ingrese su contraseña 

Iniciar sesion

No tienes cuenta? [Regístrate](#)

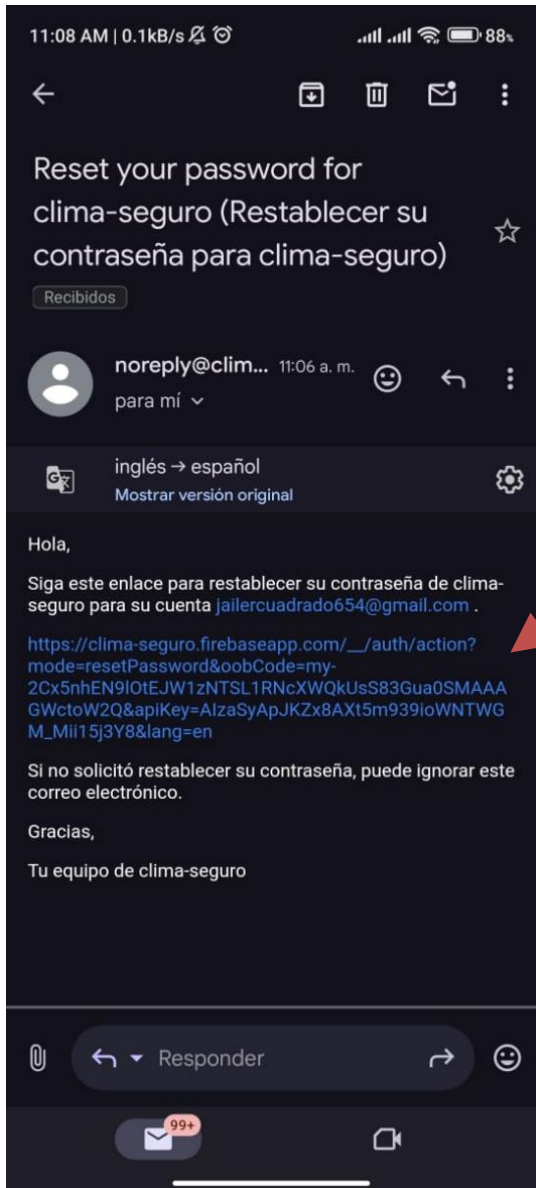
Olvidaste tu contraseña? [Recuperar contraseña](#)

Ingresar al perfil de Usuario y dar Clic en Recuperar contraseña

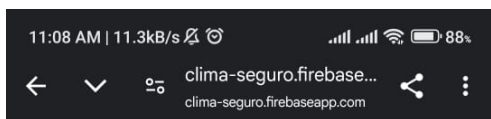


Ingresar el correo al que le va a llegar el link de recuperación

Dar clic en este botón



En el correo llegara el siguiente enlace para realizar el cambio de la clave



Se ingresa la nueva contraseña

### Reset your password

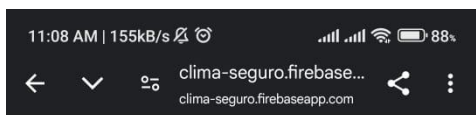
for jailercuadrado654@gmail.com

New password

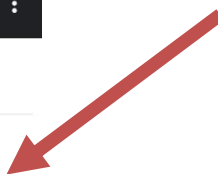


SAVE





Se muestra mensaje de cambio de contraseña



**Perfil de SST**



Inicio de sesión a perfil de SST





SST, puede visualizar todos los trabajadores activos en obra con su respectivo monitoreo de temperatura

# Monitoreo

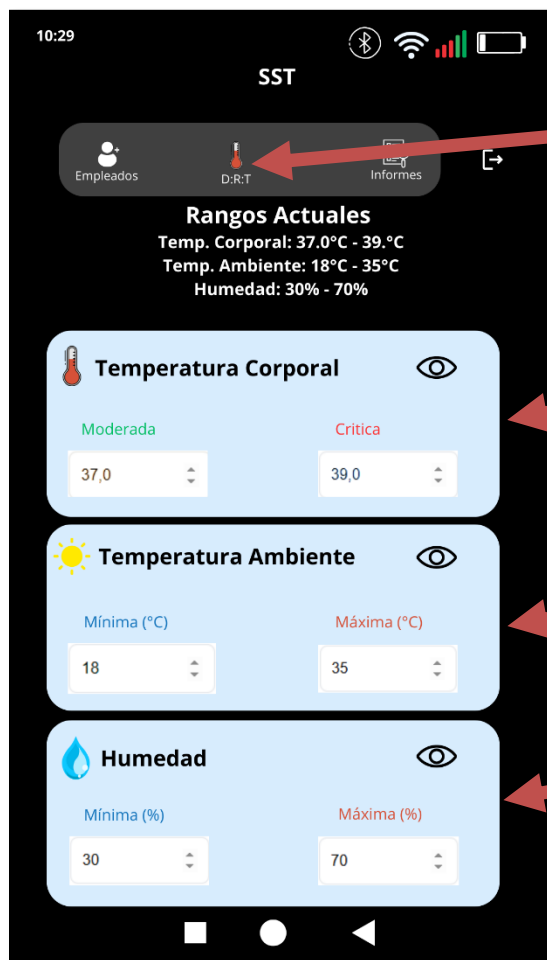


SST al ingresar tiene la vista de los estados de temperatura de cada trabajador

● Estado de temperatura Normal

● Estado de temperatura Precaución

● Estado de temperatura Peligro

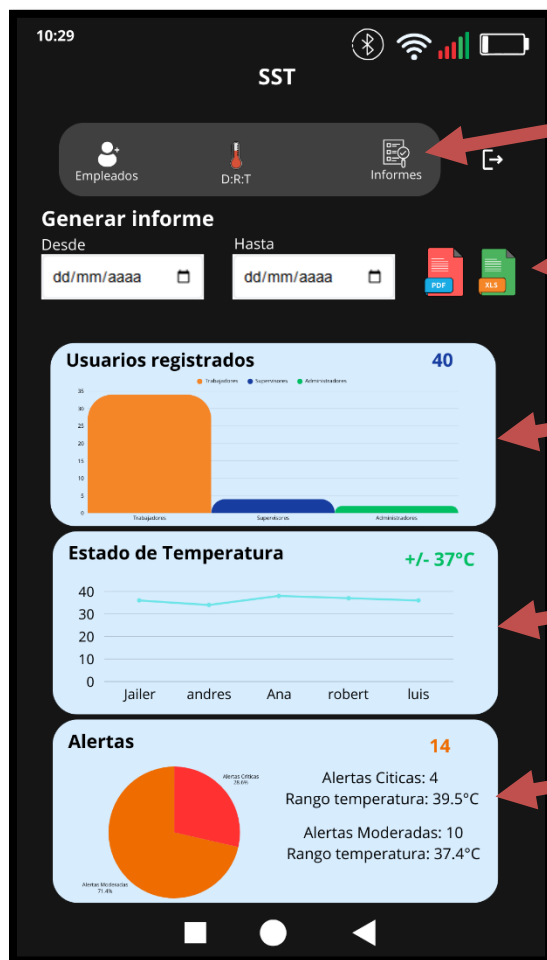


Ingresa a la opción de D:R:T(Definir rangos de temperatura), para establecer los valores mediante los cuales se activarán las alerta

Establece o modifica los valores para la temperatura corporal.

Establece o modifica los valores para la temperatura ambiente.

Establece o modifica los valores para la humedad.



Ingresa a la opción INFORMES, aquí podrá realizar consultas o informes

Opción de descarga de dicha consulta.

Informe de usuarios registrados en el sistema.

Informe de la temperatura promedio de los trabajadores activos en obra.

Informe de alertas generadas en obra.



Ingresamos al perfil del Administrador.

10:29

Sistemas de emergencias-admin

Roles Dispositivos Tips D:R:T Historial Informes Empleados

**Gestión de roles** + Nuevo Rol

**Administrador**  
Permisos: Gestión completa de usuarios, dispositivos, alertas y configuración

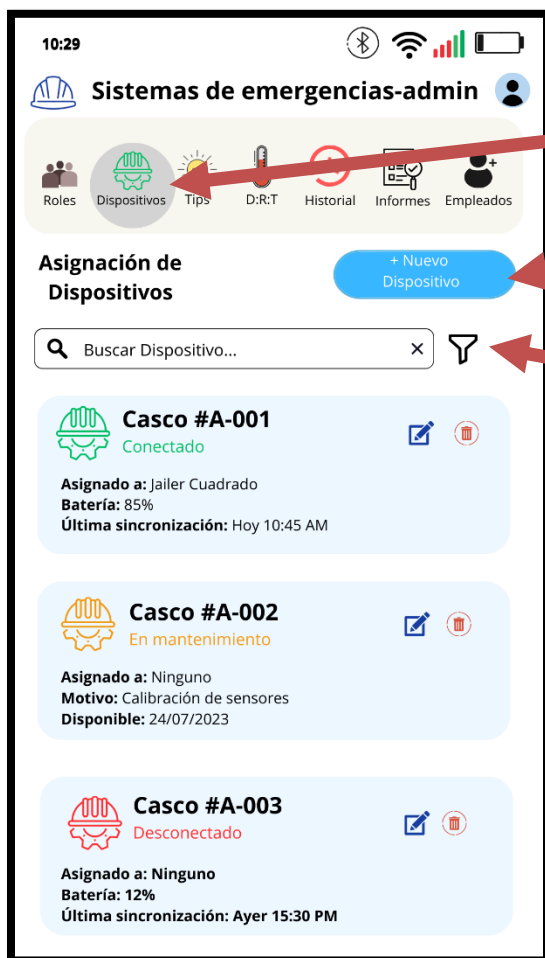
**Supervisor o SST**  
Permisos: Monitoreo de trabajadores, gestión de alertas, asignación de dispositivos

**Trabajador**  
Permisos: Ver su propio estado, recibir alertas, acceder a tips de prevención

Al ingresar al perfil por defecto inicia en la opción ROLES.

En caso de requerirlo puede crear un nuevo ROL.

Puede modificar o eliminar cada ROL en caso de requerirlo.



Opción de DISPOSITIVOS.

Puede agregar un nuevo DISPOSITIVOS.

Se puede buscar un dispositivo en específico filtrando su búsqueda por el serial del casco o el nombre del trabajador.

Para cada dispositivo se presenta la opción de modificar ya bien sea para cambiar el dispositivo de trabajador o su serial, y la opción de eliminar o deshabilitar el dispositivo.

10:29

Sistemas de emergencias-admin

Roles Dispositivos **Tips** D:R:T Historial Informes Empleados

**Gestión de Tips** + Nuevo Tip

**Hidratación constante**  
Tip #1 - Activo  
Bebe agua cada 15-20 minutos, incluso si no sientes sed. En días calurosos, necesitas al menos 2 litros de agua durante tu jornada.  
Creado: 18/06/2023 Prioridad: Alta

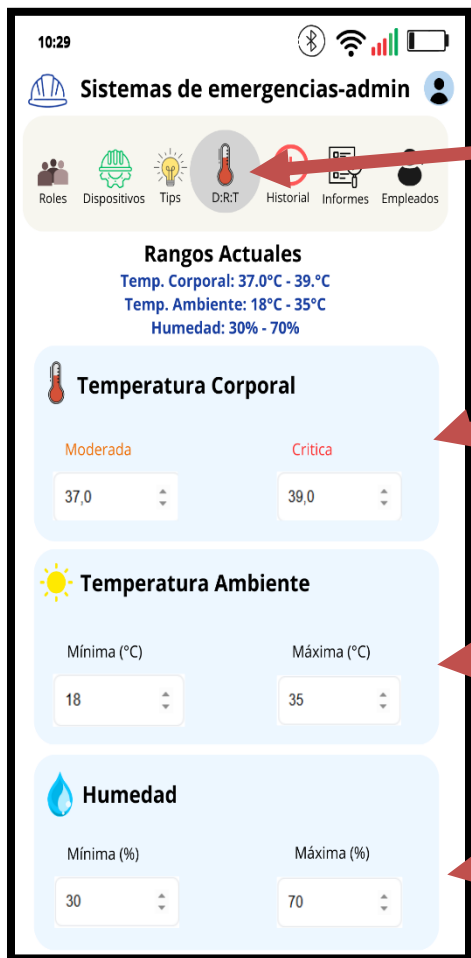
**Descansos a la sombra**  
Tip #2 - Activo  
Toma descansos de 5-10 minutos cada hora en áreas sombreadas. Permite que tu cuerpo se recupere de la exposición al calor.  
Creado: 15/06/2023 Prioridad: Media

**Ropa adecuada**  
Tip #3 - Activo  
Usa ropa ligera, de colores claros y transpirable. Protege tu cuello y cabeza del sol con una gorra o pañuelo bajo el casco.  
Creado: 20/06/2023 Prioridad: Media

Opción tips, Se pueden visualizar los diferentes tips registrados en el sistema.

Puede agregar un nuevo tips.

Para cada tip registrado, se puede modificar o eliminar.



Ingresa a la opción de D:R:T(Definir Rangos de Temperatura), para establecer los valores mediante los cuales se activaran las alertas.

Establece o modifica los valores para la temperatura corporal.

Establece o modifica los valores para la temperatura ambiente.

Establece o modifica los valores para la humedad.

10:29

Sistemas de emergencias-admin

Roles Dispositivos Tips D:R:T Historial Informes Empleados

Historial de Golpes de Calor

Buscar por nombre o fecha...

**Alerta Crítica** Crítico

Hoy, 11:23 AM

Trabajador: Jailer Cuadrado

Temp. Corporal 39.2°C

Humedad 78%

Temp. Ambiente 35.2°C

Duración 23 min

Ver más detalles

**Alerta Crítica** Moderado

Hoy, 11:23 AM

Trabajador: Luz Elena Romero

Temp. Corporal 37.8°C

Humedad 73%

Temp. Ambiente 34.6°C

Duración 16 min

Ver más detalles

**Alerta Crítica** Normal

Hoy, 11:23 AM

Trabajador: Andres Quiroga

Temp. Corporal 35.8°C

Humedad 65%

Temp. Ambiente 33.8°C

Duración 0

Ver más detalles

Ingresa a la opción Historial, aquí podrá ver diferentes alertas generadas o consultar una alerta en específico.

La consulta se puede filtrar por el nombre o la cedula del trabajador.

En el historial cada alerta se mostrará con diferentes detalles como el tipo de alerta, nombre del trabajador entre otros datos.



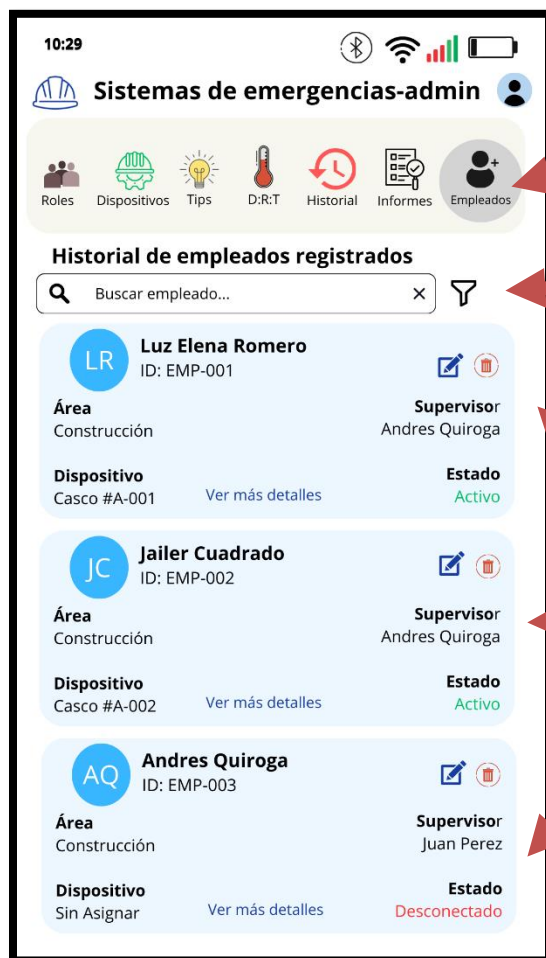
Ingresa a la opción INFORMES, aquí podrás realizar consultas o informes

Opción de descarga de dicha consulta.

Informe de usuarios registrados en el sistema.

Informe de la temperatura promedio de los trabajadores activos en obra.

Informe de alertas generadas en obra.



Opción EMPLEADO. En esta opción se pueden ver los diferentes usuarios registrados en el sistema.

Se puede realizar la búsqueda de un empleado, filtrándola bien sea por su nombre o cedula.

Para cada empleado, se muestra su respectiva información desde su cargo hasta su estado si esta activo o inactivo, de igual forma puede modificar la información de dicho empleado o eliminarlo.



**Microcontrolador ESP32:** Es un potente módulo que integra WiFi y Bluetooth, ideal para desarrollar productos de IoT. La integración de Bluetooth, Bluetooth LE y Wi-Fi permite una amplia gama de aplicaciones, el uso de WiFi permite una comunicación de mediano alcance y conectarse a una red LAN y a través de un Modem Router conexión a Internet, mientras que el Bluetooth nos permite conectarse directamente a otro dispositivo como un celular. (Sigma Electrónica, s.f.) (Sigma Electronica, s.f.)

#### ESPECIFICACIONES TECNICAS

- Microcontrolador: Xtensa LX6 de 32 bits, dual-core (o single-core)
- Tensión de funcionamiento: 3.3V (máx. 3.6V)
- Pines de E/S digitales: Hasta 34
- Pines de E/S digitales con PWM: Hasta 16 canales
- Pines de entrada analógica: 15 (12-bit SAR ADC)
- SPI/I2C/I2S/UART: 4/2/2/3
- Memoria flash: Hasta 16 MB (externa), 448 KB (interna)
- SRAM: 520 KB
- Velocidad de reloj: 160 MHz o 240 MHz
- Wi-Fi: Sí, 802.11 b/g/n
- Bluetooth: Sí, v4.2 BR/EDR y BLE
- Sensor táctil: 10 sensores capacitivos



- **Sensor de temperatura corporal (MLX90614):**

Es un sensor de temperatura infrarrojo sin contacto, ideal para medir la temperatura de objetos a distancia. Utiliza una termopila MEMS que detecta la radiación infrarroja y convierte la señal en una lectura digital. El sensor incluye un filtro de ruido, conversor A/D de 17 bits y un DSP, lo que mejora la precisión. Para utilizarlo con Arduino o ESP32, es necesario identificar su dirección I2C, lo que puede hacerse con un I2C Scanner. En comparación con el MLX90614ESF-BAA, este modelo tiene mayor precisión y mejor compensación térmica, lo que lo hace más fiable en entornos variables. (Electronica plug and play, s.f.)

#### ESPECIFICACIONES TECNICAS

- Voltaje de operación: 3.3V-5V DC
- Protocolo de comunicación: SMBUS (subconjunto del I2C)
- Rango de temperatura ambiental: -40°C hasta +85°C
- Rango de temperatura de objeto: -70°C hasta +380°C
- Precisión:  $\pm 0.2^{\circ}\text{C}$
- Ángulo de visión: 5° (FOV)
- ADC interno de 17 bits
- Procesador digital de señal interno (DSP)
- Regulador de voltaje 3.3V en placa
- Resistencias Pull-up a VIN en placa
- No necesita componentes adicionales
- Peso: 5 gramos



- **Sensor de temperatura y humedad (SHT31):**

Este sensor permite realizar mediciones muy precisas a bajo costo. Le ofrece una mejor calidad que los sensores DHT11 y DHT22, especialmente mejor resolución, mayor precisión y compactación de empaque. Se emplea en aplicaciones usadas en temperatura, tales como control automático, aire acondicionado, monitoreo en agricultura, entre otros.

Sencillez de su integración, tanto en componentes software como hardware. En este sentido, en lo que respecta a software, se dispone de librerías para Arduino con soporte para el protocolo I2C. En lo que respecta a hardware, para su uso solo son necesarios 4 cables: dos de alimentación y dos de datos I2C.

(Macatronica Electronica y tecnologia desde 2014, s.f.)

ESPECIFICACIONES TECNICAS:

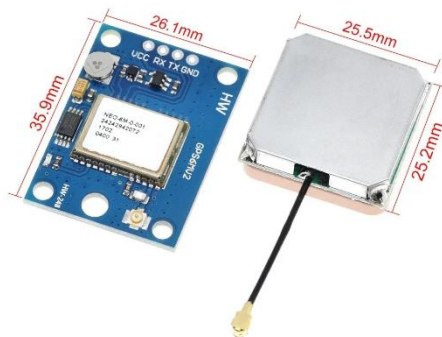
- Voltaje de Operación: 2.4V a 5.5V DC
- Interfaz de comunicación: I2C
- Dirección I2C: 0x44
- Rango de trabajo Temperatura: -40° a 125°C
- Resolución Temperatura: 0.015°C
- Precisión Temperatura: 0.2°C
- Rango de trabajo Humedad: 0 a 100% RH
- Resolución HR: 0.01 %RH
- Precisión HR: 2% RH
- Tiempos de muestreo rápidos



- **Módulo GPS (NEO-6M):** Diseñado con base al módulo U-Blox 6 un motor de posicionamiento de 50 canales cuenta con un tiempo hasta la primera reparación (TTFF) de menos de 1 segundo, ofreciendo gran rendimiento y estabilidad. Capaz de realizar búsquedas masivas de espacio de tiempo / frecuencia en paralelo, adquisición, con hasta 2 millones de adquisiciones correlacionadas permitiendo encontrar satélites al instante. Incluye una memoria EEPROM para poder almacenar datos de configuración del módulo. (Sigma electronica, s.f.)

#### ESPECIFICACIONES TECNICAS

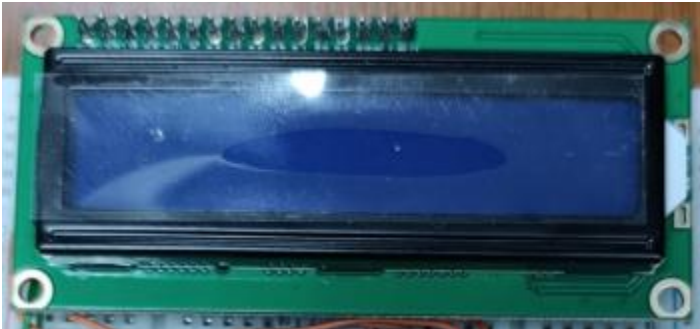
- Voltaje de alimentación: 3.3 V ~ 5.0 V.
- Consumo de corriente: ~ 100 mA.
- Recepción: 50 Canales; GPS L1 frequency, C/A Code, SBAS: WAAS, EGNOS, MSAS.
- Tiempo de enganche promedio: 1 s ~ 27 s.
- Precisión de Posicionamiento: 2.5 MCEP.
- Protocolo de comunicación: NMEA, UBX binario.
- Velocidad de comunicación: 9600 baudios.



### **Pantalla LCD JHD162A**

Permite visualizar en tiempo real los datos, como lecturas de sensores y alertas de seguridad. Se conecta al microcontrolador ESP32 y funciona mediante instrucciones que controlan qué texto mostrar. Gracias a su simplicidad, ofrece una interfaz directa y útil para el usuario, incluso sin acceso a la aplicación móvil. (Electronicos Caldas , s.f.)

- 16 caracteres x 2 líneas
- Puede mostrar letras, números, caracteres especiales, y hasta 8 caracteres creados por el usuario
- Backlight de LED color verde
- Interface paralela. Puede operar en modo de 8 bits, o de 4 bits para ahorrar pines del microcontrolador
- Posee controlador Hitachi HD44780 o compatible on-board
- Voltaje de alimentación nominal: 5 V



### Buzzer piezoeléctrico KY-012

El buzzer piezoeléctrico se utiliza para emitir alertas sonoras cuando el sistema detecta condiciones climáticas peligrosas o anómalas. Funciona aplicando una señal eléctrica que hace vibrar un material interno, generando sonido. Está conectado al ESP32, que lo activa según los parámetros definidos por los sensores. Esta señal auditiva complementa la pantalla LCD y la app móvil, asegurando que el usuario reciba notificaciones inmediatas incluso sin mirar el dispositivo. (TIENDA DE COMPONENTES ELECTRÓNICOS, s.f.)

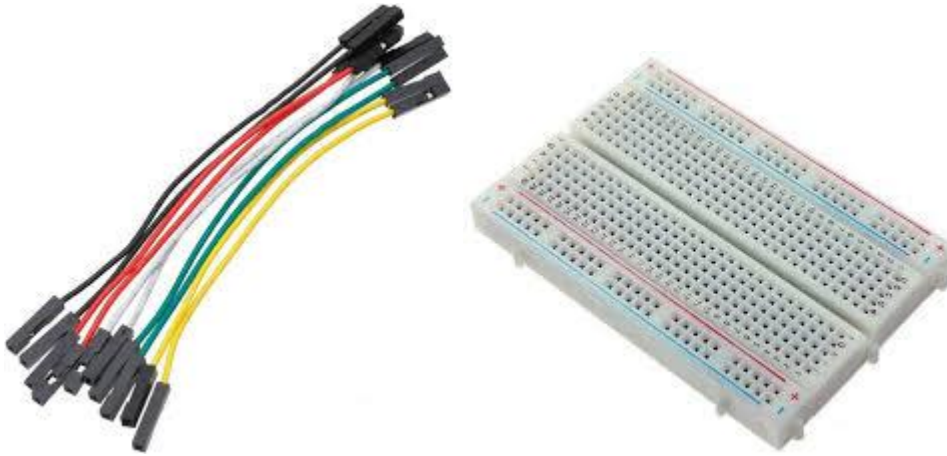
#### ESPECIFICACIONES TECNICAS

- **Tipo:** Buzzer piezoeléctrico activo.
- **Voltaje de operación:** 3.3V – 5V (compatible con ESP32).
- **Frecuencia de tono:** ~2.5 kHz.
- **Nivel de sonido:** mínimo 85 dB.
- **Corriente máxima:** 30 mA.
- **Funcionamiento:** Emite sonido automáticamente al recibir voltaje en el pin de señal, sin necesidad de generar una señal PWM.



### Cables de conexión y protoboard

En el sistema se utilizan cables de puente tipo Dupont, que permiten conectar fácilmente los componentes electrónicos al microcontrolador ESP32 sin necesidad de soldadura. Estos cables vienen en diferentes colores para facilitar la identificación de conexiones. (Electronilab, s.f.)



### Entorno de Desarrollo

#### SOFTWARE

- Software necesario:
- Flutter SDK 3.22
- Dart SDK integrado
- Visual Studio Code
- Android Studio
- IDE Arduino, para cargar el programa.

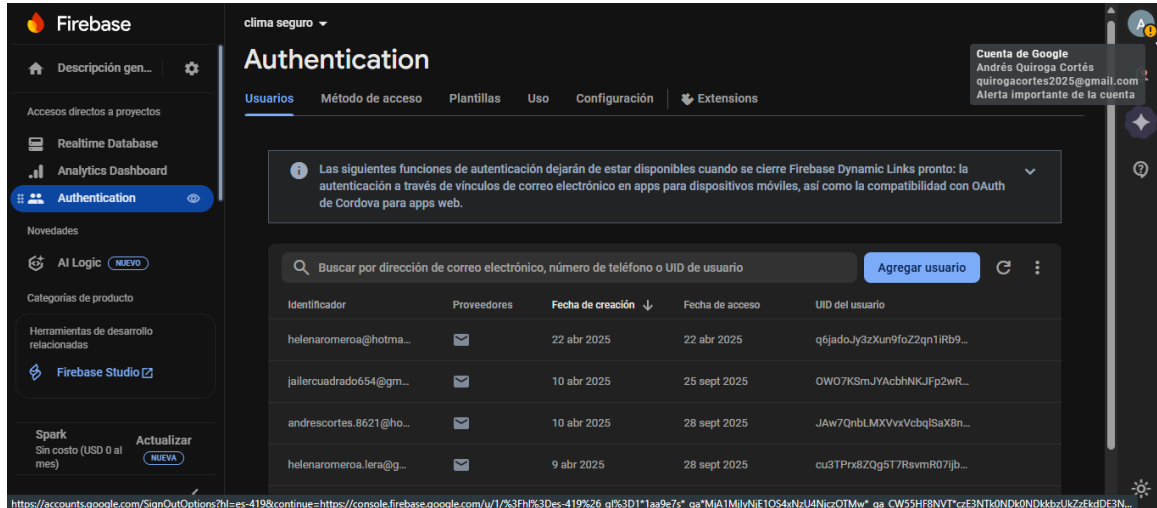
#### Entorno de desarrollo:

Visual Studio Code, por su ligereza, extensibilidad y compatibilidad con Flutter.

- **Framework:**  
Flutter SDK (versión 3.22 o superior), que permite desarrollar aplicaciones móviles nativas para Android e iOS desde una sola base de código.
- **Lenguaje de programación:**  
Dart, el lenguaje oficial de Flutter, orientado a objetos y optimizado para interfaces reactivas.

➤ **Backend en la nube:**

Firestore, utilizado para autenticación, almacenamiento de datos y configuración remota.



The screenshot shows the Firebase Authentication console for a project named "clima seguro". The left sidebar contains navigation options like "Descripción gen...", "Realtime Database", "Analytics Dashboard", and "Authentication". The main content area is titled "Authentication" and includes a search bar and a table of users.

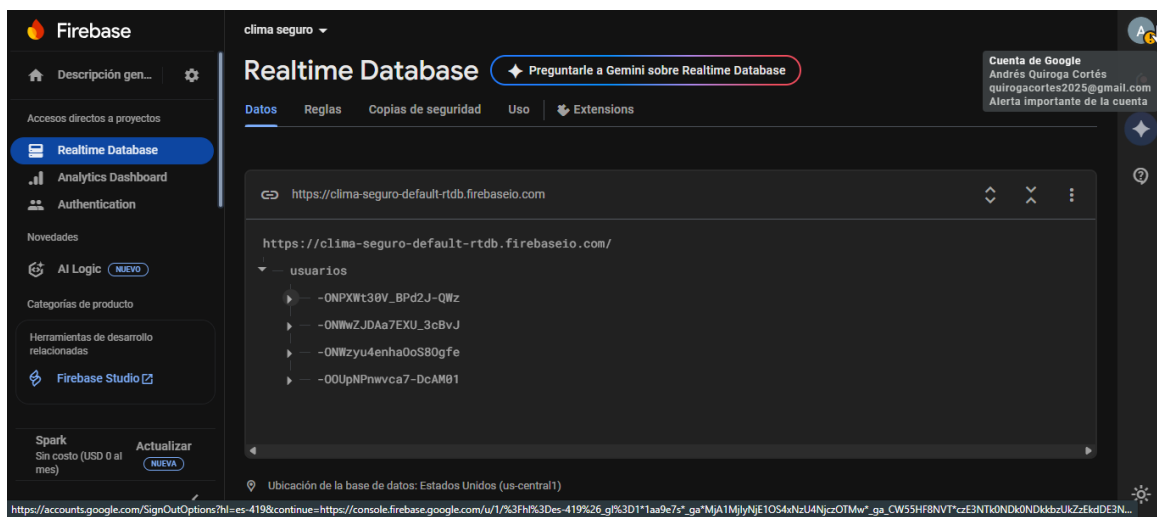
**Authentication**

Usuarios Método de acceso Plantillas Uso Configuración Extensiones

Las siguientes funciones de autenticación dejarán de estar disponibles cuando se cierre Firebase Dynamic Links pronto: la autenticación a través de vínculos de correo electrónico en apps para dispositivos móviles, así como la compatibilidad con OAuth de Cordova para apps web.

Buscar por dirección de correo electrónico, número de teléfono o UID de usuario [Agregar usuario](#)

Identificador	Proveedores	Fecha de creación ↓	Fecha de acceso	UID del usuario
helenaromeroa@hotmail...	✉	22 abr 2025	22 abr 2025	q6jadoJy3Zxun9foZ2qn1Rb9...
jallercuadrado654@gm...	✉	10 abr 2025	25 sept 2025	OW07KSmJYAcbhNKJFp2wR...
andrescortes.8621@ho...	✉	10 abr 2025	28 sept 2025	JAw7QnbLMXVvxVcbqSaX8n...
helenaromeroa.lera@g...	✉	9 abr 2025	28 sept 2025	cu3TPrx8ZQg5T7RsvmR07jb...



The screenshot shows the Firebase Realtime Database console for the same project. The left sidebar is similar to the previous screenshot. The main content area is titled "Realtime Database" and shows a tree view of the database structure.

**Realtime Database** [Preguntarle a Gemini sobre Realtime Database](#)

Datos Reglas Copias de seguridad Uso Extensiones

https://clima-seguro-default-rtdb.firebaseio.com

```

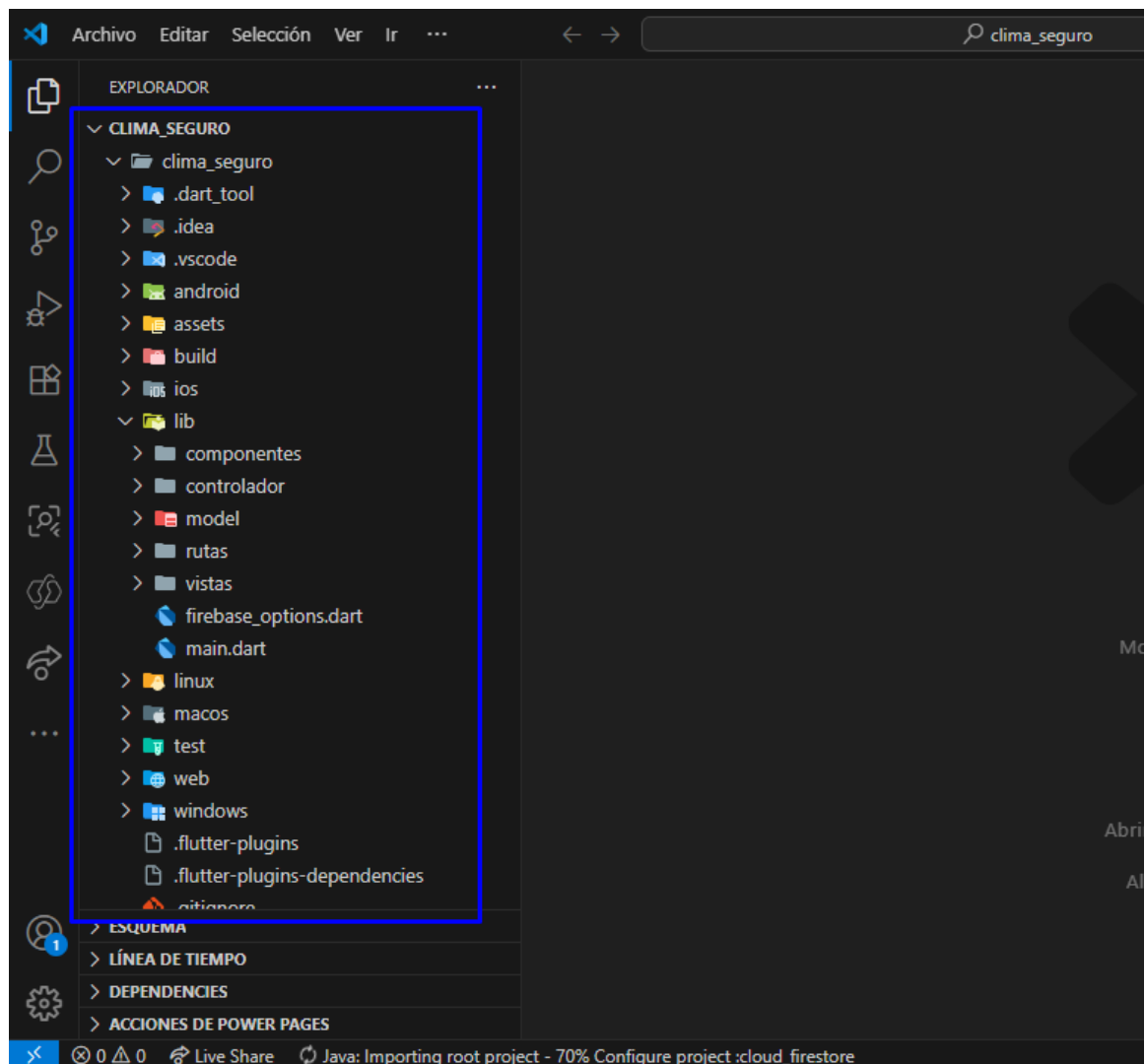
https://clima-seguro-default-rtdb.firebaseio.com/
├── usuarios
│   ├── -ONPXWt38V_BPdZJ-QWz
│   ├── -ONWwZJDAa7EXU_3cBvJ
│   ├── -ONWzyu4enha0o580gfe
│   └── -00UpNPnwca7-DcAM81

```

Ubicación de la base de datos: Estados Unidos (us-central1)

El proyecto está organizado en carpetas que separan la lógica de negocio, la interfaz de usuario y los servicios:

- /lib/screens: contiene las pantallas principales de la app.
- /lib/services: incluye la lógica de conexión con Firebase y Bluetooth.
- /lib/models: define las estructuras de datos.
- /lib/widgets: componentes reutilizables de UI.
- firebase\_options.dart: archivo generado automáticamente para configurar Firebase.



Estas son las librerías clave que permiten el funcionamiento del sistema:

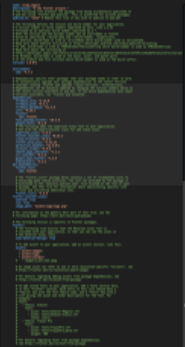
<b>Dependencia</b>	<b>Versión</b>	<b>Función principal</b>
firebase_core	^3.13.0	Inicializa Firebase en la app.
firebase_auth	^5.3.1	Autenticación de usuarios.
firebase_database	^11.3.5	Base de datos en tiempo real (Realtime Database).
cloud_firestore	^5.6.6	Base de datos NoSQL para almacenamiento estructurado.
flutter_bluetooth_serial	^0.4.0	Comunicación con el ESP32 vía Bluetooth.
permission_handler	^12.0.0+1	Gestión de permisos del sistema operativo.
audioplayers	^5.2.1	Reproducción de sonidos de alerta.
intl	^0.18.1	Formateo de fechas, horas y textos.
url_launcher	^6.2.5	Apertura de enlaces externos desde la app.
google_maps_flutter	^2.6.0	Integración de mapas para geolocalización.
carousel_slider	^5.0.0	Carruseles de imágenes o contenido.
smooth_page_indicator	^1.2.1	Indicadores visuales para navegación entre páginas.
font_awesome_flutter	^10.5.0	Iconografía avanzada para mejorar la interfaz.
google_fonts	^6.1.0	Tipografías personalizadas de Google.
cupertino_icons	^1.0.8	Iconos estilo iOS para compatibilidad visual.
flutter_launcher_icons	^0.14.3	Personalización del ícono de la app



## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

```
! pubspec.yaml
26 # consider running `flutter pub upgrade --major-versions`. Alternatively,
27 # dependencies can be manually updated by changing the version numbers below to
28 # the latest version available on pub.dev. To see which dependencies have newer
29 # versions available, run `flutter pub outdated`.
30 dependencies:
31   firebase_core: ^3.13.0
32   firebase_auth: ^5.3.1
33   firebase_database: ^11.3.5
34   intl: ^0.18.1
35   flutter:
36     sdk: flutter
37   font_awesome_flutter: ^10.5.0
38   google_fonts: ^6.1.0
39   # The following adds the Cupertino Icons font to your application.
40   # Use with the CupertinoIcons class for iOS style icons.
41   cupertino_icons: ^1.0.8
42   flutter_launcher_icons: ^0.14.3
43   cloud_firestore: ^5.6.6
44   flutter_blue_plus: ^1.31.15
45   permission_handler: ^12.0.0+1
46   carousel_slider: ^5.0.0
47   smooth_page_indicator: ^1.2.1
48   url_launcher: ^6.2.5
49   google_maps_flutter: ^2.6.0
50   audioplayers: ^5.2.1
51 dev_dependencies:
52   flutter_test:
53     sdk: flutter
54
```



## TRABAJO DE GRADO

### Opción Investigación o Proyecto de Grado

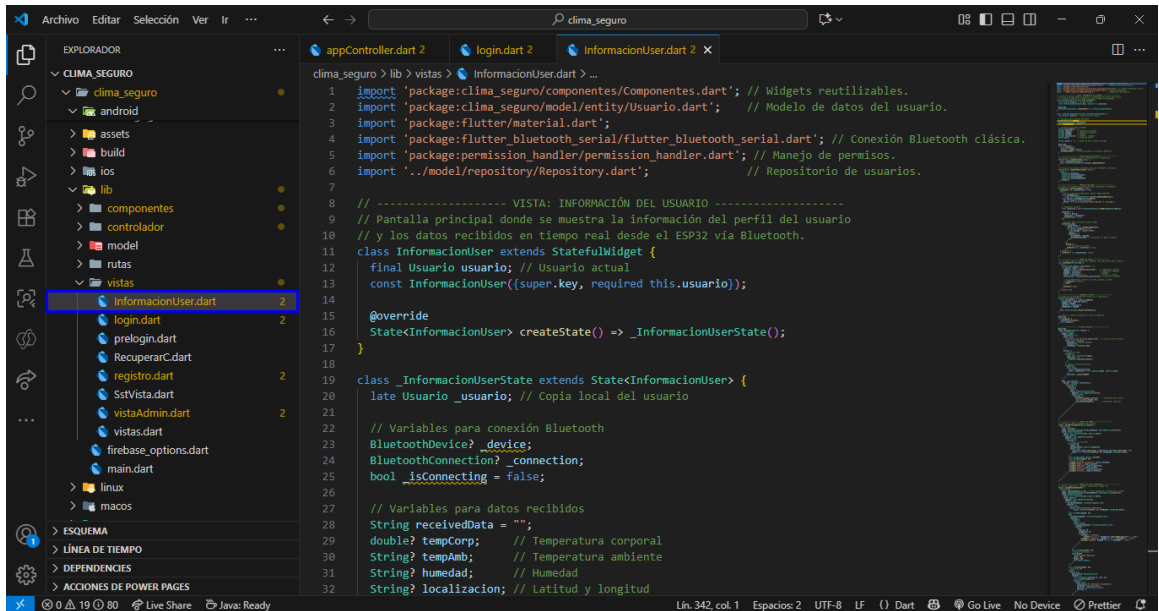
#### Instalación y Configuración

1. Descargar e instalar Flutter SDK y configurar las variables de entorno.
2. Instalar Android Studio, para configurar emuladores.
3. Usar Visual Studio Code como editor de código.
4. Clonar el repositorio del proyecto y Ejecutar en la terminal del proyecto 'flutter pub get' para instalar dependencias, luego ejecutar: Flutter run para correr el proyecto
5. Configurar Firebase y agregar google-services.json en Android.
6. Cargar el programa en Arduino UNO desde el IDE de Arduino.
7. Emparejar el dispositivo móvil con el casco vía Bluetooth.
8. Ejecutar en la terminal el siguiente código flutter build apk para generar una APK y poder visualizar la aplicación en un dispositivo móvil con 'flutter run'.

#### Estructura del Proyecto

Esta carpeta **lib/**: contiene el código principal.

- vistas/: pantallas como Login, Registro, Vistaadmin, Sstvista.



```
EXPLORADOR
├── CLIMA SEGURO
│   ├── clima_seguro
│   │   ├── android
│   │   ├── assets
│   │   ├── build
│   │   ├── ios
│   │   ├── lib
│   │   │   ├── componentes
│   │   │   ├── controlador
│   │   │   ├── model
│   │   │   ├── rutas
│   │   │   └── vistas
│   │   │       ├── informacionUser.dart
│   │   │       ├── login.dart
│   │   │       ├── prelogin.dart
│   │   │       ├── RecuperarC.dart
│   │   │       ├── registro.dart
│   │   │       ├── SstVista.dart
│   │   │       ├── vistaAdmin.dart
│   │   │       ├── vistas.dart
│   │   │       ├── firebase_options.dart
│   │   │       └── main.dart
│   │   ├── linux
│   │   └── macos
│   ├── ESQUEMA
│   ├── LÍNEA DE TIEMPO
│   ├── DEPENDENCIAS
│   └── ACCIONES DE POWER PAGES
└── ...

appController.dart 2
login.dart 2
InformacionUser.dart 2

clima_seguro > lib > vistas > InformacionUser.dart > ...
1 import 'package:clima_seguro/componentes/Componentes.dart'; // Widgets reutilizables.
2 import 'package:clima_seguro/model/entity/Usuario.dart'; // Modelo de datos del usuario.
3 import 'package:flutter/material.dart';
4 import 'package:flutter_bluetooth_serial/flutter_bluetooth_serial.dart'; // Conexión Bluetooth clásica.
5 import 'package:permission_handler/permission_handler.dart'; // Manejo de permisos.
6 import '../model/repository/Repository.dart'; // Repositorio de usuarios.
7
8 // ----- VISTA: INFORMACIÓN DEL USUARIO -----
9 // Pantalla principal donde se muestra la información del perfil del usuario
10 // y los datos recibidos en tiempo real desde el ESP32 vía Bluetooth.
11 class InformacionUser extends StatefulWidget {
12   final Usuario usuario; // Usuario actual
13   const InformacionUser({super.key, required this.usuario});
14
15   @override
16   State<InformacionUser> createState() => _InformacionUserState();
17 }
18
19 class _InformacionUserState extends State<InformacionUser> {
20   late Usuario _usuario; // Copia local del usuario
21
22   // Variables para conexión Bluetooth
23   BluetoothDevice? _device;
24   BluetoothConnection? _connection;
25   bool _isConnecting = false;
26
27   // Variables para datos recibidos
28   String receivedData = "";
29   double? tempCorp; // Temperatura corporal
30   String? tempAmb; // Temperatura ambiente
31   String? humedad; // Humedad
32   String? localizacion; // Latitud y longitud
```

➤ model/: entidades y repositorios (ejemplo: Usuario.dart).

The image displays two screenshots of an IDE (VS Code) showing Dart code for a user entity and repository. The top screenshot shows the `Usuario.dart` file, and the bottom screenshot shows the `UsuariosRepo.dart` file.

**Top Screenshot: `Usuario.dart`**

```

class Usuario {
  // Temperatura ambiente en el área donde se encuentra el usuario.
  final String? temAmbiente;

  // Rol del usuario dentro de la aplicación (ej: "trabajador", "supervisor").
  final String? rol;

  // Constructor de la clase Usuario.
  // Algunos campos son obligatorios, otros son opcionales o nulos.
  Usuario({
    this.id,
    required this.nombre,
    required this.apellidos,
    required this.cedula,
    required this.fechaNac,
    required this.telefono,
    required this.direccion,
    required this.correo,
    this.temperatura,
    this.humedad,
    this.ubicacion,
    this.temAmbiente,
    this.rol,
  });

  // Método que crea una copia del objeto actual con posibilidad de modificar
  // algunos atributos sin alterar el resto (muy útil para actualizaciones parciales).
  Usuario copyWith({
    String? id,
    String? nombre,
    String? apellidos,
    String? cedula
  });
}

```

**Bottom Screenshot: `UsuariosRepo.dart`**

```

class UsuariosRepo {
  // Referencia a la base de datos en la rama "usuarios"
  // dentro de Firebase Realtime Database.
  final DatabaseReference _dbRef = FirebaseDatabase.instanceFor(
    app: Firebase.app(),
    databaseURL: "https://clima-seguro-default-rtadb.firebaseio.com",
  ).ref("usuarios");

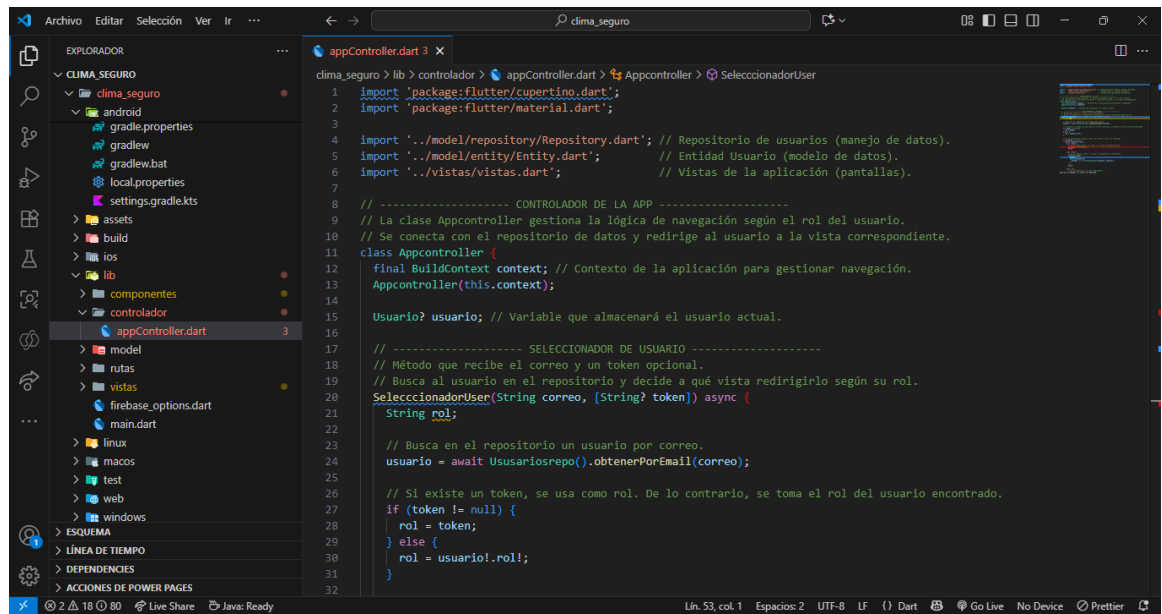
  // Obtiene un stream (flujo) en tiempo real de la lista de usuarios.
  // Se actualiza automáticamente cada vez que ocurre un cambio en la BD.
  Stream<List<Usuario>> get usuariosStream {
    return _dbRef.onValue.map((DatabaseEvent event) {
      final data = event.snapshot.value as Map<dynamic, dynamic>;
      if (data == null) return [];

      // Convierte cada entrada del mapa en un objeto Usuario
      return data.entries.map((entry) {
        return Usuario.fromMap({
          ..Map<String, dynamic>.from(entry.value),
          'id': entry.key, // Asigna la clave como ID del usuario
        }); // Usuario.fromMap
      }).toList();
    });
  }

  // Edita un usuario existente en la BD.
  // Requiere que el usuario tenga un ID válido.
  Future<void> editar(Usuario usuario) async {
    print("Editando usuario con ID: ${usuario.id}");
    try {
      if (usuario.id == null) {
        throw Exception("El ID del usuario no puede ser nulo.");
      }
    }
  }
}

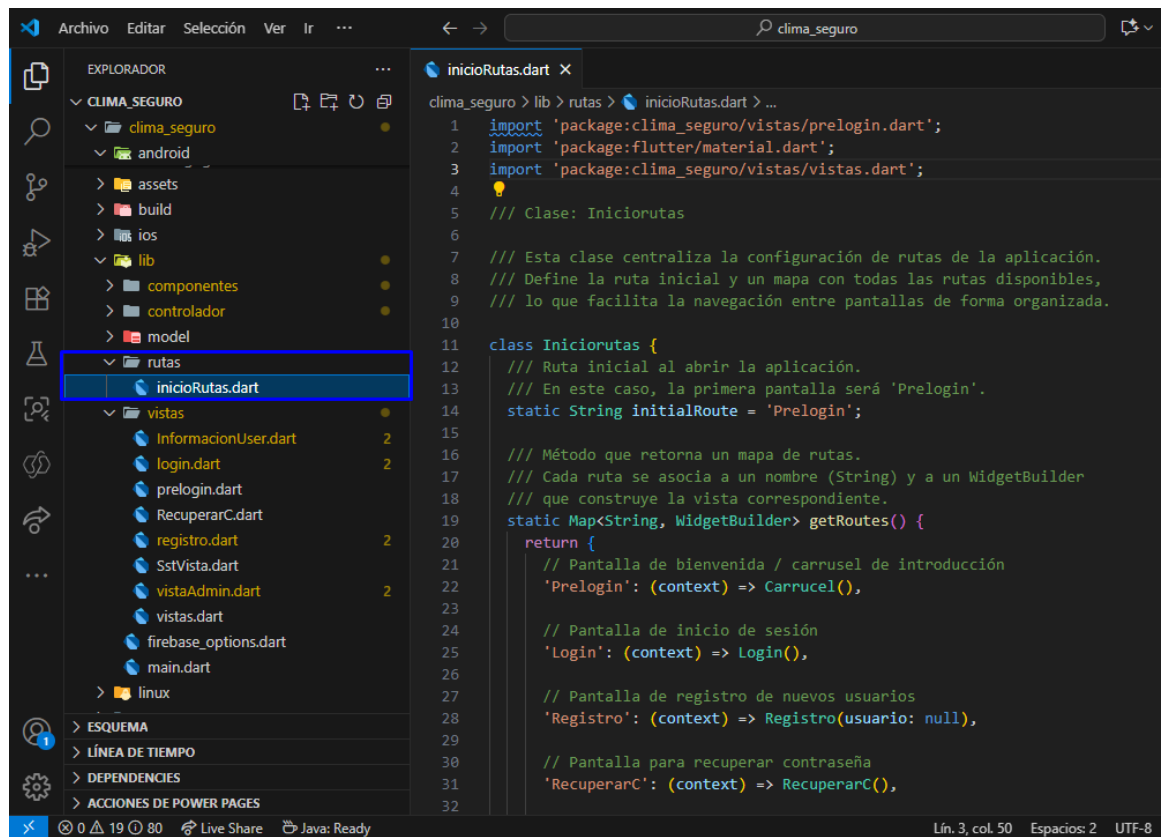
```

## ➤ controlador/: lógica de negocio y controladores.



```
1 import 'package:flutter/cupertino.dart';
2 import 'package:flutter/material.dart';
3
4 import '../model/repository/Repository.dart'; // Repositorio de usuarios (manejo de datos).
5 import '../model/entity/Entity.dart'; // Entidad Usuario (modelo de datos).
6 import '../vistas/vistas.dart'; // Vistas de la aplicación (pantallas).
7
8 // ----- CONTROLADOR DE LA APP -----
9 // La clase Appcontroller gestiona la lógica de navegación según el rol del usuario.
10 // Se conecta con el repositorio de datos y redirige al usuario a la vista correspondiente.
11 class Appcontroller {
12   final BuildContext context; // Contexto de la aplicación para gestionar navegación.
13   Appcontroller(this.context);
14
15   Usuario? usuario; // Variable que almacenará el usuario actual.
16
17 // ----- SELECCIONADOR DE USUARIO -----
18 // Método que recibe el correo y un token opcional.
19 // Busca al usuario en el repositorio y decide a qué vista redirigirlo según su rol.
20   SeleccionadorUser(String correo, [String? token]) async {
21     String rol;
22
23     // Busca en el repositorio un usuario por correo.
24     usuario = await Usuariosrepo().obtenerPorEmail(correo);
25
26     // Si existe un token, se usa como rol. De lo contrario, se toma el rol del usuario encontrado.
27     if (token != null) {
28       rol = token;
29     } else {
30       rol = usuario!.rol!;
31     }
32   }
33 }
```

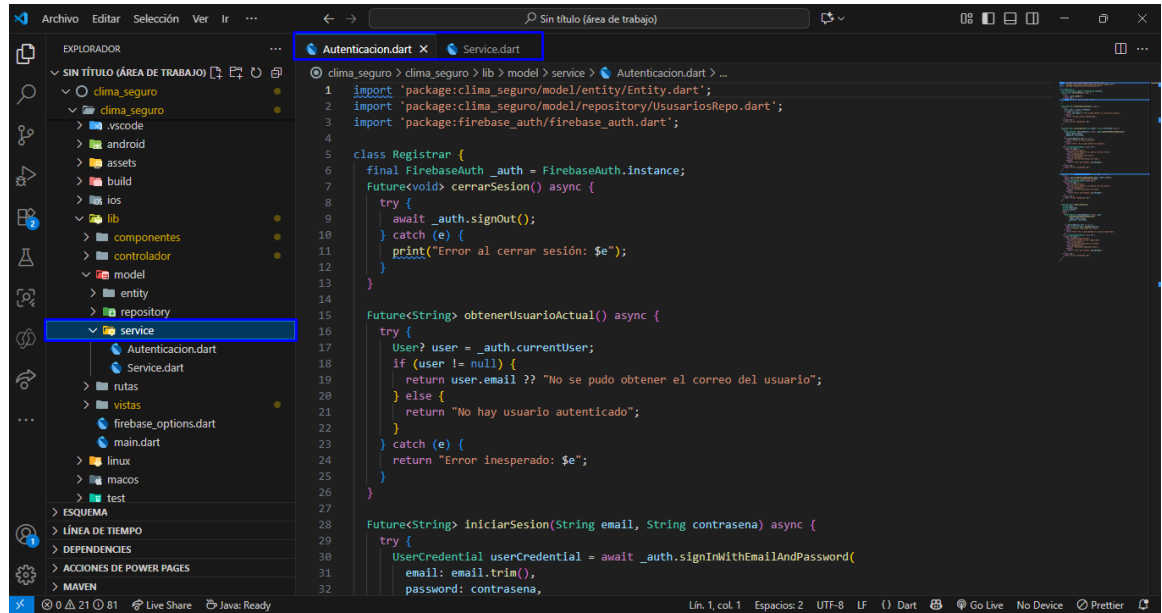
➤ rutas/: manejo de navegación entre pantalla



```
Archivo Editar Selección Ver Ir ... clima_seguro
EXPLORADOR
CLIMA_SEGURO
  clima_seguro
    android
    assets
    build
    ios
    lib
      componentes
      controlador
      model
      rutas
        inicioRutas.dart
      vistas
        InformacionUser.dart
        login.dart
        prelogin.dart
        RecuperarC.dart
        registro.dart
        SstVista.dart
        vistaAdmin.dart
        vistas.dart
        firebase_options.dart
        main.dart
    linux
  ESQUEMA
  LÍNEA DE TIEMPO
  DEPENDENCIAS
  ACCIONES DE POWER PAGES

inicioRutas.dart
clima_seguro > lib > rutas > inicioRutas.dart > ...
1 import 'package:clima_seguro/vistas/prelogin.dart';
2 import 'package:flutter/material.dart';
3 import 'package:clima_seguro/vistas/vistas.dart';
4
5 /// Clase: Iniciorutas
6
7 /// Esta clase centraliza la configuración de rutas de la aplicación.
8 /// Define la ruta inicial y un mapa con todas las rutas disponibles,
9 /// lo que facilita la navegación entre pantallas de forma organizada.
10
11 class Iniciorutas {
12   /// Ruta inicial al abrir la aplicación.
13   /// En este caso, la primera pantalla será 'Prelogin'.
14   static String initialRoute = 'Prelogin';
15
16   /// Método que retorna un mapa de rutas.
17   /// Cada ruta se asocia a un nombre (String) y a un WidgetBuilder
18   /// que construye la vista correspondiente.
19   static Map<String, WidgetBuilder> getRoutes() {
20     return {
21       // Pantalla de bienvenida / carrusel de introducción
22       'Prelogin': (context) => Carrusel(),
23
24       // Pantalla de inicio de sesión
25       'Login': (context) => Login(),
26
27       // Pantalla de registro de nuevos usuarios
28       'Registro': (context) => Registro(usuario: null),
29
30       // Pantalla para recuperar contraseña
31       'RecuperarC': (context) => RecuperarC(),
32     };
33   }
34 }
```

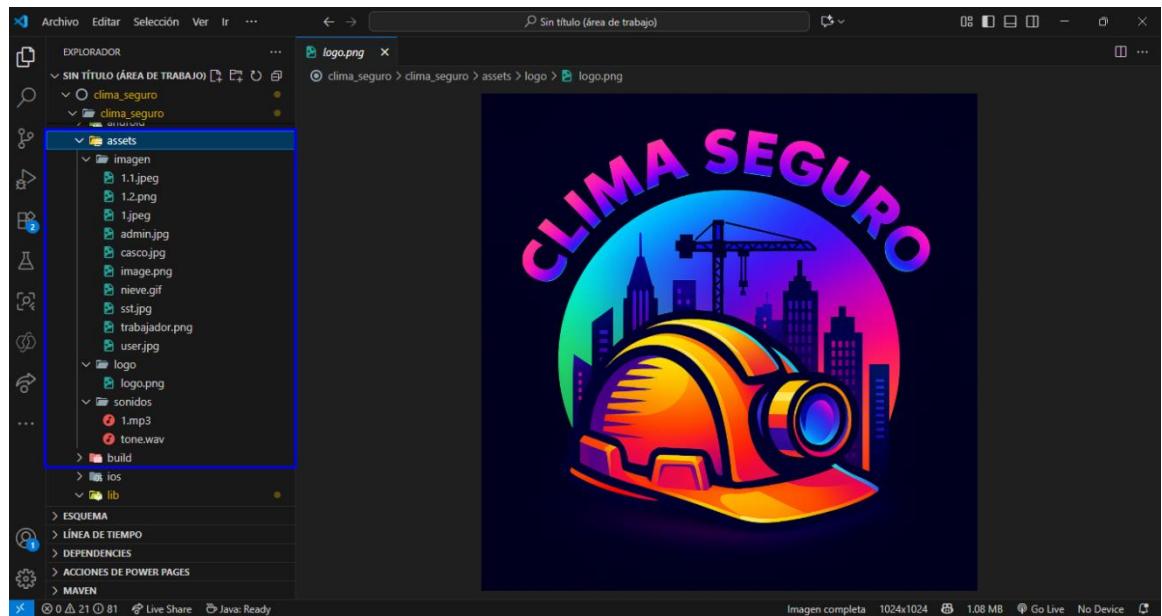
- services/: servicios de conexión con hardware y Firebase.



The screenshot shows a code editor with the following Dart code:

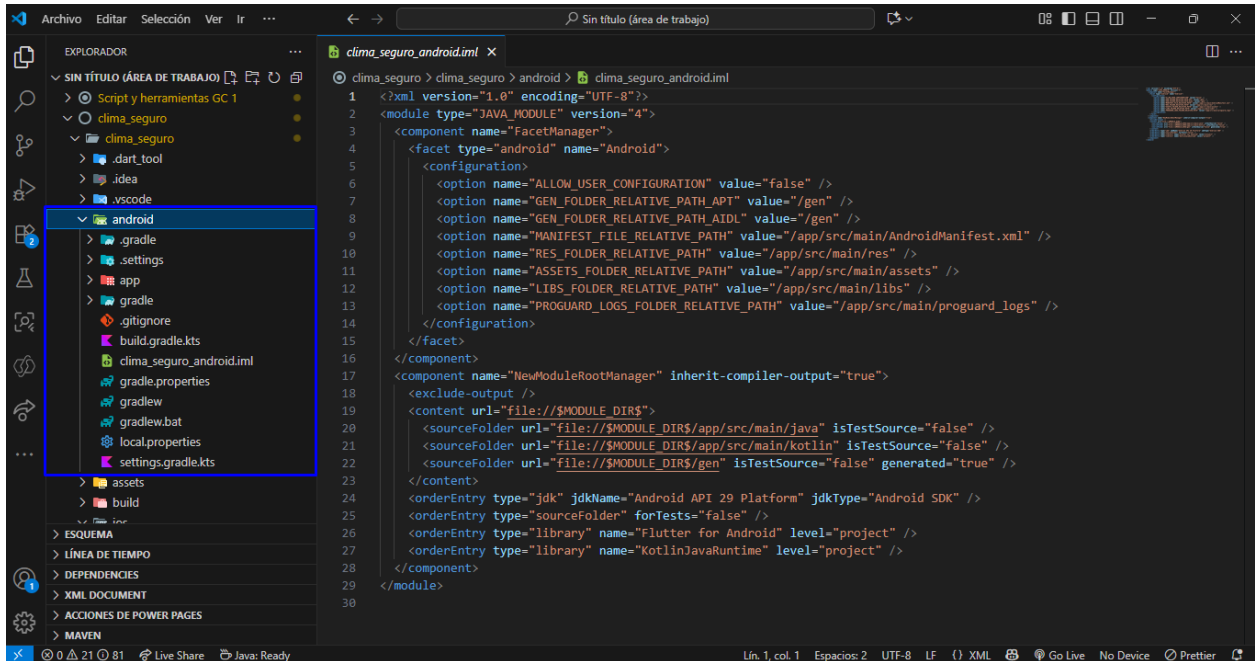
```
1 import 'package:clima_seguro/model/entity/Entity.dart';
2 import 'package:clima_seguro/model/repository/UsuariosRepo.dart';
3 import 'package:firebase_auth/firebase_auth.dart';
4
5 class Registrar {
6   final FirebaseAuth _auth = FirebaseAuth.instance;
7   Future<void> cerrarSesion() async {
8     try {
9       await _auth.signOut();
10    } catch (e) {
11      print("Error al cerrar sesión: $e");
12    }
13  }
14
15  Future<String> obtenerUsuarioActual() async {
16    try {
17      User? user = _auth.currentUser;
18      if (user != null) {
19        return user.email ?? "No se pudo obtener el correo del usuario";
20      } else {
21        return "No hay usuario autenticado";
22      }
23    } catch (e) {
24      return "Error inesperado: $e";
25    }
26  }
27
28  Future<String> iniciarSesion(String email, String contrasena) async {
29    try {
30      UserCredential userCredential = await _auth.signInWithEmailAndPassword(
31        email: email.trim(),
32        password: contrasena,
```

➤ assets/: imágenes y recursos estáticos.

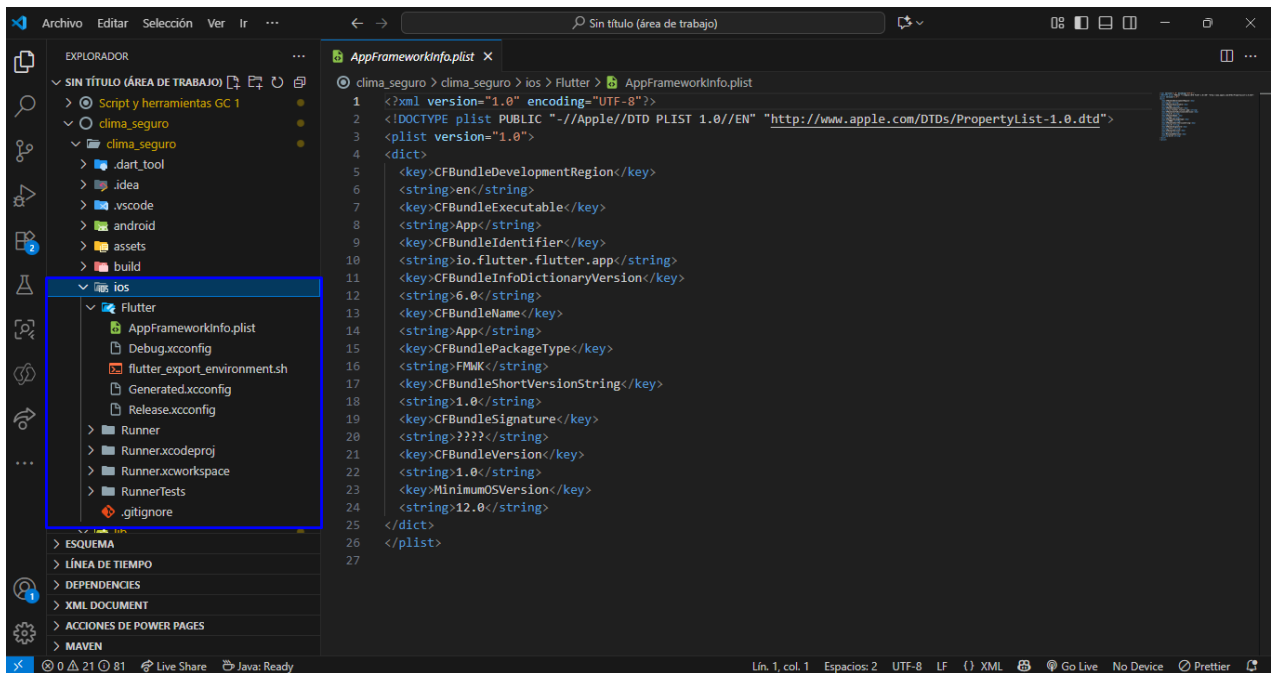


- android/ y ios/: configuración para cada plataforma.

## Aandroid



## ios



## Mantenimiento y Actualización

El sistema puede actualizarse mediante la modificación de las dependencias en `pubspec.yaml` y la ejecución de `'flutter pub upgrade'` para el software, se debe de verificar que se esté utilizando la última versión de Flutter SDK y mantener sincronizados los servicios de autenticación de Firebase. Los parámetros de sensores pueden ajustarse en el código de Arduino. En caso de fallas de hardware, los sensores deben reemplazarse, para ello se recomienda tener un inventario básico de repuestos. El sistema implementa roles de usuario (admin, user, sst). Se recomienda cifrar contraseñas en Firebase y realizar copias de seguridad periódicas de la base de datos. Es fundamental documentar cualquier cambio realizado en el código.

## 29. Fase de Pruebas

El objetivo principal de esta fase de pruebas es validar que el sistema cumpla con los requisitos funcionales y requisitos no funcionales previamente definidos en la fase de análisis y diseño, Además de estos con las pruebas también se busca asegurar que todas las funciones como la lectura de datos desde los sensores del casco, la generación de las alertas y la gestión de usuarios entre otras, se ejecuten de forma correcta. De igual forma además de validar las funcionalidades, con las pruebas se espera evaluar la usabilidad, rendimiento y seguridad del sistema.

### 1. Pruebas de Revisión de Código

Escenario	Requisit o Asociad o	Resultado Esperado	Resultado Obtenido	¿Pasa ?	Observaciones
Validar nombres claros en clases y métodos	RF1	Los nombres deben ser representativos y consistentes.	Los nombres de clases y widgets son claros y siguen las convenciones de Flutter.	✓	Se recomienda mantener la convención <code>lowerCamelCase</code> para variables y métodos.
Revisar manejo de excepciones	RNF2	El sistema debe manejar excepciones con <code>try/catch</code> y mostrar	El código maneja errores con <code>try/catch</code> , pero algunos	✗	Agregar <code>SnackBar</code> o <code>AlertDialog</code> para mensajes

		mensajes claros.	mensajes no se muestran al usuario.		de error visibles.
Verificar modularidad	RF3	Separar la lógica de negocio y la interfaz de usuario.	El proyecto mantiene archivos independientes para vistas, controladores y servicios.	✓	Correcto uso del patrón MVC simplificado.
Confirmar cierre de streams y controladores	RNF3	Los controladores deben cerrarse al salir de la vista.	Algunos controladores se cierran correctamente, otros no.	✗	Agregar dispose() en todos los controladores.

## 2. Pruebas Funcionales

Escenario	Requisito Asociado	Resultado Esperado	Resultado Obtenido	¿Pasa?	Observaciones
Registrar nuevo usuario (trabajador o SST)	RF1	El usuario se registra y se guarda en Firebase.	El registro se completa correctamente.	✓	Se recomienda agregar validación de correo duplicado.
Lectura de sensores desde el casco	RF2	Los datos se reciben y almacenan en Firebase.	Las lecturas llegan correctamente con leve retraso.	✓	Latencia promedio de 1.2 segundos.
Generar alerta de temperatura alta	RF3	Se genera alerta naranja o roja según los umbrales definidos.	Alertas generadas y notificadas en tiempo real.	✓	Funcionalidad validada correctamente.

Consulta de historial de alertas	RF4	Mostrar alertas por usuario y fecha.	Historial visible y exportable.	✓	Correcto funcionamiento.
Desconexión del casco	RNF1	Mostrar notificación al perder conexión.	No muestra alerta de desconexión.	✗	Agregar notificación de pérdida de conexión.

### 3.Pruebas de Seguridad

Escenario	Requisito Asociado	Resultado Esperado	Resultado Obtenido	¿Pasa ?	Observaciones
Intentar acceder sin autenticación	RNF3	Denegar acceso a pantallas restringidas.	FirestoreAuth bloquea acceso.	✓	Correcta implementación de seguridad.
Modificar datos directamente desde Firestore	RNF1	Solo usuarios autorizados pueden escribir.	Reglas Firestore impiden modificación directa.	✓	Reglas bien configuradas.
Validar datos antes de guardar	RNF4	No se guardan datos nulos o inválidos.	Se permiten algunos valores nulos.	✗	Agregar validación en el cliente y backend.
Intentar acceder con token caducado	RNF3	Debe cerrar sesión automáticamente.	El sistema redirige a login.	✓	Correcta gestión de sesión expirada.

#### 4. Pruebas de Estrés, Usabilidad, Aceptación y Regresión

Tipo de Prueba	Historia de Usuario	Requisito Asociado	Escenario Asociado	Resultado Esperado	¿Pasa?	Observaciones
Estrés	Como usuario, quiero enviar múltiples lecturas sin fallos.	RNF2	Simular 500 lecturas seguidas.	Sistema mantiene estabilidad.	✓	Retraso leve en Firebase bajo carga alta.
Usabilidad	Como usuario, quiero recibir alertas claras.	RNF5	Simular evento de alerta.	Notificación visual y sonora generada.	✓	Alertas efectivas.
Aceptación	Como SST, quiero consultar reportes históricos.	RF4	Exportar alertas filtradas.	Archivo generado con éxito.	✓	Listo para entrega final.
Regresión	Como tester, quiero validar registro tras actualización.	RF1	Repetir prueba de registro.	Funciona correctamente tras actualización.	✓	Sin errores de regresión.

## 5. Herramientas Utilizadas

Tipo de Prueba	Herramienta	Descripción
Funcionales	Flutter Test	Framework para pruebas unitarias e integración en Flutter.
Seguridad	Firebase Rules Simulator	Simulador para validar reglas de Firestore.
Estrés	Apache JMeter	Herramienta para simular múltiples usuarios simultáneos.
Usabilidad	OBS Studio	Permite grabar sesiones de uso para análisis de experiencia.
Aceptación	Postman	Pruebas de endpoints de Firebase Functions.

## 30. Resultados y Discusión

Desarrollar este trabajo de grado permitió diseñar y poner en funcionamiento un software y su respectivo hardware que está pensado para medir la temperatura corporal mediante un casco inteligente, los principales éxitos alcanzados con este desarrollo fueron:

### Resultados:

- El monitoreo y procesamiento en tiempo real de los datos que son captados desde el trabajador hacia el sensor.
- El diseño intuitivo de una interfaz que permitiera a cualquier persona u empresa acceder y utilizar de manera clara y precisa la aplicación.
- La integración de funcionalidades que permiten alertar y notificar cuando exista algún tipo de riesgo para la salud de quien se requiere monitorear.
- La viabilidad que se obtuvo al realizar las respectivas pruebas del prototipo, mostrando la estabilidad y objetividad de este en tiempo y escenarios reales sin perder o distorsionar la información.

**Aspectos para considerar y estudios futuros:**

- Realizar más pruebas en diferentes tipos de escenarios tales como actividades al aire libre y recreación, eventos deportivos, seminarios de salud, etc., con el fin de verificar y poner a prueba el funcionamiento del software y hardware en diferentes campos y perfiles de usuario.
- La aplicación puede llegar a ser más personalizada incluyendo alertas personalizadas e incluyendo las necesidades específicas que cada persona quiera monitorear o inclusive condicionar el software a ciertos factores climáticos o a personas con determinados estados de salud.
- Se podría mejorar la forma y el tiempo de almacenamiento podría ser por módulos que permitan un mejor registro y consulta a futuro para eventos clínicos o simplemente como consulta preventiva para el usuario o la empresa.
- Explorar la posibilidad de integrar la aplicación con otros softwares existentes en el ámbito laboral, médico o deportivo que permitan completar su uso y así ampliar su alcance y utilidad para las organizaciones.
- A futuro lograr una mayor durabilidad tanto en el tiempo de vida útil del sistema como en el material en que está construido el casco.

**Conclusiones**

- Diseñar e implementar este proyecto de grado nos permitió integrar funcionalidades para brindar a las empresas un aporte que permita contribuir al bienestar de los empleados, aplicar el monitoreo en tiempo real nos permitirá ser una opción confiable y segura para las personas que lo usen, además de dejar sentadas bases para posteriores desarrollos de dispositivos de protección.
- Este casco inteligente demuestra que es posible fusionar en un solo elemento temas tan relevantes como la salud, tecnología, innovación, desarrollo y medición, este software diseñado abre la posibilidad de creación y mejoramiento de elementos para la medición corporal que pueden ser aplicados en los ámbitos laborales, deportivos y médicos.
- La propuesta de este software de desarrollar un sistema de monitoreo basado en un casco inteligente fue alcanzado a satisfacción ya que se logró indagar, procesar y poner en marcha el artefacto que permite medir la temperatura corporal e integrar sensores para detectar y prevenir posibles accidentes laborales.

- Si bien se logró cumplir con los objetivos propuestos al inicio del proyecto, se detectaron oportunidades de mejora para posteriores versiones tales como la personalización del casco dependiendo de la industria o sector donde se desee utilizar y su integración con los diferentes dispositivos fijos y móviles.
- El Desarrollo de este proyecto responde a una necesidad actual de prevención y control del riesgo para cuidar la salud de los empleados, especialmente donde el control de temperatura juega un papel primordial para el desarrollo de sus funciones, este casco puede convertirse en una herramienta esencial para el área de salud ocupacional en el trabajo como método de ayuda, control y monitoreo diario del estado de los colaboradores.

**TRABAJO DE GRADO**  
**Opción Investigación o Proyecto de Grado**

**Referencias**

- American Academy of Ophthalmology*. (25 de Junio de 2024). Obtenido de Cáncer ocular:  
<https://www.aao.org/salud-ocular/enfermedades/cancer-de-ojo>
- Cleveland Clinic*. (2025). Obtenido de Fotoqueratitis:  
<https://my.clevelandclinic.org/health/diseases/15763-photokeratitis>
- Clinica Baviera*. (s.f.). Obtenido de Pterigión o carnosidad en los ojos: ¿qué es y por qué sale?: <https://www.clinicabaviera.com/pterigion-tratamiento>
- Clinica Baviera*. (s.f.). Obtenido de Cataratas: <https://www.clinicabaviera.com/blog/que-efectos-tiene-el-sol-sobre-los-ojos/#:~:text=Cataratas,ultravioleta%20puede%20acelerar%20este%20proceso.>
- Diseño y desarrollo web*. (s.f.). Obtenido de DEVELOVER: <https://develover.mx/mapa-de-navegacion-web-que-es-y-como-implementarlo/>
- Electronica plug and play*. (s.f.). Obtenido de Sensor de temperatura infrarrojo MLX90614: <https://www.electronicaplugandplay.com/sensores-y-transductores/product/822-mlx90614esf-baa-infrared-sensor>
- Electronicos Caldas*. (s.f.). Obtenido de Pantalla LCD JHD162A: <https://www.electronicoscaldas.com/es/displays-lcd-alfanumericos-y-graficos/130-display-lcd-16x2-backlight-verde.html?srsltid=AfmBOorr7KvGQk5ULPuibqcRThk6Y6lnDQgsxkBEeM-4ndBxXuLSN9GA>
- Electronilab*. (s.f.). Obtenido de Cables conectores para protoboard macho-macho: [https://electronilab.co/tienda/65-cables-conectores-para-protoboard-macho-macho/?srsltid=AfmBOoqe5ulxkzEsduqrc9FxcbQCT10Sw0UKjQxLKVXtwbtRJ9v11\\_ne](https://electronilab.co/tienda/65-cables-conectores-para-protoboard-macho-macho/?srsltid=AfmBOoqe5ulxkzEsduqrc9FxcbQCT10Sw0UKjQxLKVXtwbtRJ9v11_ne)
- Especificación de requisitos de software*. (6 de Marzo de 2024). Obtenido de Wikipedia: [https://es.wikipedia.org/wiki/Especificaci%C3%B3n\\_de\\_requisitos\\_de\\_software](https://es.wikipedia.org/wiki/Especificaci%C3%B3n_de_requisitos_de_software)
- Evaluando Software*. (2025). Obtenido de Qué es la geolocalización y cómo funciona: <https://www.evaluandosoftware.com/bpm/la-geolocalizacion-funciona/>
- Fernandez, H. A. (Julio de 2013). *Rev. Vinculos, vol. 6, n.º 1, pp.* Obtenido de PROCESOS DE INGENIERÍA DE SOFTWARE: <http://revistas.udistrital.edu.co:8080/index.php/vinculos/article/view/4141>
- Garcia, E. M. (28 de junio de 2024). *Prolaboral*. Obtenido de Exposición a la radiación solar en el trabajo: [https://www.prolaboral.com/es/blog/exposicion-a-la-radiacion-solar-en-el-trabajo.html?srsltid=AfmBOop04yh5UJpGhMQV\\_hTRsg890DvL4DMPAdTFJ7\\_w3ZsZ9NQFM7FS](https://www.prolaboral.com/es/blog/exposicion-a-la-radiacion-solar-en-el-trabajo.html?srsltid=AfmBOop04yh5UJpGhMQV_hTRsg890DvL4DMPAdTFJ7_w3ZsZ9NQFM7FS)
- Garzas, J. (3 de Noviembre de 2022). *Javiergarzas.com*. Obtenido de Los 4 riesgos de la creación de un producto y los roles que los gestionan: <https://www.javiergarzas.com/2022/11/los-4-riesgos-de-un-producto-y-los-roles.html>

- Gobierno de Mexico*. (16 de Abril de 2025). Obtenido de ¿Qué es una ola u onda de calor?: <https://www.gob.mx/smn/es/articulos/que-es-una-ola-u-onda-de-calor?idiom=es>
- Granada, U. d. (2025). *Decsai*. Obtenido de Departamento de Ciencias de la Computación e Inteligencia Artificial: <https://decsai.ugr.es/>
- IFRC*. (2025). Obtenido de Olas de calor: <https://www.ifrc.org/es/nuestro-trabajo/desastres-clima-y-crisis/que-es-desastre/olas-calor>
- Macatronica Electronica y tecnologia desde 2014*. (s.f.). Obtenido de SENSOR DE TEMPERATURA Y HUMEDAD SHT31: [https://www.mactronica.com.co/sensor-de-temperatura-y-humedad-sht31?srsltid=AfmBOoqPxe7LM\\_HrMTOinQtQofBgxAqYfPXtZrAhX4bMQPIIcNEC6WoE](https://www.mactronica.com.co/sensor-de-temperatura-y-humedad-sht31?srsltid=AfmBOoqPxe7LM_HrMTOinQtQofBgxAqYfPXtZrAhX4bMQPIIcNEC6WoE)
- Martins, J. (15 de Febrero de 2025). Obtenido de Asana: <https://asana.com/es/resources/what-is-scrum>
- Minambiente*. (2022). Obtenido de <https://www.minambiente.gov.co/planeacion-y-seguimiento/sistema-de-gestion-de-seguridad-y-salud-en-el-trabajo/>
- Naylamp Mechatronics*. (2023). Obtenido de Sensor de temperatura y humedad relativa SHT31: <https://naylampmechatronics.com/sensores-temperatura-y-humedad/363-sensor-de-temperatura-y-humedad-relativa-sht31.html>
- Naylamp Mechatronics*. (2023). Obtenido de Sensor de temperatura infrarrojo MLX90614 (GY-906): <https://naylampmechatronics.com/sensores-temperatura-y-humedad/330-sensor-de-temperatura-infrarrojo-mlx90614-gy-906.html>
- Obagi medical*. (19 de Noviembre de 2024). Obtenido de Hiperpigmentación: por qué ocurre y cómo ayuda la hidroquinona: [https://www.obagi.com/blogs/news/hyperpigmentation-why-it-occurs-and-how-hydroquinone-helps?srsltid=AfmBOoqGGTv5YCyK8ioQlKiqRK6V8-ta\\_3NKPjvhcbwVduKKSq7x0T8m](https://www.obagi.com/blogs/news/hyperpigmentation-why-it-occurs-and-how-hydroquinone-helps?srsltid=AfmBOoqGGTv5YCyK8ioQlKiqRK6V8-ta_3NKPjvhcbwVduKKSq7x0T8m)
- Open Academy*. (21 de Diciembre de 2020). Obtenido de Santander: <https://www.santanderopenacademy.com/es/blog/metodologias-desarrollo-software.html>
- Organización Mundial de la Salud*. (28 de Mayo de 2024). Obtenido de Calor y salud: <https://www.who.int/es/news-room/fact-sheets/detail/climate-change-heat-and-health>
- Park, B. M. (19 de Julio de 2024). *Isdin*. Obtenido de Qué es el fotoenvejecimiento y cómo se refleja en tu piel: <https://www.isdin.com/es/blog/efectos-del-fotoenvejecimiento-en-la-piel/>
- Pilas RECARGABLES 3.7 V 2.5 Ah Amp horas*. (s.f.). Obtenido de Batería recargable LiPo 3.7V, 2500mAh: [https://co.ebay.com/b/Rechargeable-Batteries-3-7-V-2-5-Ah-Amp-Hours/48619/bn\\_95406777](https://co.ebay.com/b/Rechargeable-Batteries-3-7-V-2-5-Ah-Amp-Hours/48619/bn_95406777)
- PUBLICACIONES SEMANA S.A.* (17 de Septiembre de 2024). Obtenido de Estas son las regiones de Colombia donde está haciendo más calor; las altas temperaturas superan los 40 grados: <https://www.semana.com/nacion/articulo/estas-son-las>

- regiones-de-colombia-donde-esta-haciendo-mas-calor-las-altas-temperaturas-superan-los-40-grados/202446/#google\_vignette
- Quiroz, A. (27 de Junio de 2025). *B2chat*. Obtenido de ¿Qué es una aplicación móvil y para qué sirve?: <https://www.b2chat.io/blog/marketing/aplicacion-movil-que-para-que-sirve/>
- Sigma electronica*. (s.f.). Obtenido de Módulo GPS NEO-6M V2: <https://www.sigmaelectronica.net/producto/gy-neo-6m-v2/>
- Sigma Electronica*. (s.f.). Obtenido de Microcontrolador ESP32: <https://www.sigmaelectronica.net/producto/abx00070/>
- Sigmaelectronica*. (2025). Obtenido de Módulo GPS con EEPROM Neo-6. : <https://www.sigmaelectronica.net/producto/gy-neo-6m-v2/>
- Sigmaelectronica*. (2025). Obtenido de ESP-32: <https://www.sigmaelectronica.net/producto/esp-32/>
- Smith, G. M. (04 de Marzo de 2025). *Dewesoft*. Obtenido de ¿Qué es un sensor y qué hace?: <https://dewesoft.com/es/blog/que-es-un-sensor>
- Softplan*. (15 de Enero de 2024). Obtenido de Sistema de monitoreo: Supervisión y control para una gestión eficiente: <https://sectorpublico.softplan.com.br/blog/sistema-de-monitoreo/#:~:text=Un%20sistema%20de%20monitoreo%20es,as%C3%AD%20aplicar%20los%20ajustes%20necesarios.>
- TIENDA DE COMPONENTES ELECTRÓNICOS*. (s.f.). Obtenido de Buzzer Pasivo : <https://uelectronics.com/producto/buzzer-pasivo-ky-006/>
- Universidad Europea*. (06 de Mayo de 2024). Obtenido de Microcontrolador: ¿Qué es y para qué sirve?: <https://universidadeuropea.com/blog/que-es-microcontrolador/>
- Worldwide Cancer Research*. (26 de Mayo de 2025). Obtenido de ¿Qué es el cáncer de piel?: <https://www.worldwidecancerresearch.org/cancer-and-cancer-research-explained/cancer-myths-and-questions/how-does-the-sun-cause-cancer/#:~:text=da%C3%B1en%20las%20c%C3%A9lulas,-%C2%BFC%C3%B3mo%20el%20sol%20causa%20c%C3%A1ncer?,protector%20solar%20con%20FP>