



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE TECNOLOGÍAS DE LA INFORMACIÓN,
TELECOMUNICACIONES E INDUSTRIAL
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS

TEMA:

**“SOFTWARE DE CONTROL METEOROLÓGICO PARA LA DETECCIÓN
DE HELADAS Y EMISIÓN DE ALERTAS TEMPRANAS EN EL
GEOPORTAL DEL HONORABLE GOBIERNO PROVINCIAL DE
TUNGURAHUA”**

Proyecto de Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Paul Alberto Panata Buñay

TUTOR: Ing. Franklin Mayorga, Mg.

Ambato – Ecuador

Enero 2020

APROBACIÓN DEL TUTOR

En mi calidad de tutor del Trabajo de Investigación sobre el tema: “SOFTWARE DE CONTROL METEOROLÓGICO PARA LA DETECCIÓN DE HELADAS Y EMISIÓN DE ALERTAS TEMPRANAS EN EL GEOPORTAL DEL HONORABLE GOBIERNO PROVINCIAL DE TUNGURAHUA”, del señor Paul Alberto Panata Buñay, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los tramites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, enero 2020

EL TUTOR

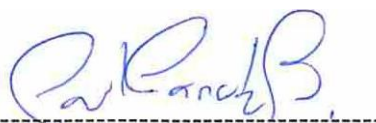


Ing. Franklin Mayorga, Mg.

AUTORÍA

El presente Proyecto de Investigación titulado: “SOFTWARE DE CONTROL METEOROLÓGICO PARA LA DETECCIÓN DE HELADAS Y EMISIÓN DE ALERTAS TEMPRANAS EN EL GEOPORTAL DEL HONORABLE GOBIERNO PROVINCIAL DE TUNGURAHUA”, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, enero 2020



Paul Alberto Panata Buñay

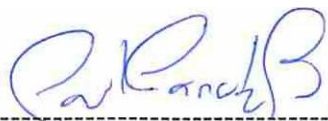
CC: 1804592549

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los Derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, enero 2020



Paul Alberto Panata Buñay

CC: 1804592549

APROBACIÓN DEL TRIBUNAL DE GRADO

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Carlos Israel Núñez Miranda Mg. e Ing. Hernán Fabricio Naranjo Ávalos Mg., revisó y aprobó el Informe Final del Proyecto de Investigación titulado “SOFTWARE DE CONTROL METEOROLÓGICO PARA LA DETECCIÓN DE HELADAS Y EMISIÓN DE ALERTAS TEMPRANAS EN EL GEOPORTAL DEL HONORABLE GOBIERNO PROVINCIAL DE TUNGURAHUA”, presentado por el señor Paul Alberto Panata Buñay, de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, enero 2020



.....

Ing. Elsa Pilar Urrutia Urrutia Mg.

PRESIDENTA ENCARGADA DEL TRIBUNAL



.....

Ing. Carlos Israel Núñez Miranda Mg
DOCENTE CALIFICADOR



.....

Ing. Hernán Fabricio Naranjo Ávalos Mg.
DOCENTE CALIFICADOR

DEDICATORIA:

Dedico el presente trabajo a mis padres ya que sin su ayuda nunca hubiera podido culminar este ciclo en mi vida.

Gracias viejos, por su apoyo.

Paul Alberto Panata Buñay

AGRADECIMIENTO:

A Dios, por permitirme estar un día más con vida y poder disfrutar cada momento en la vida terrenal.

A mis padres, por su amor, apoyo incondicional y sobre todo sus enseñanzas que me han ayudado a ser una persona de bien.

A mis compañeros y compañeras, quienes me han acompañado durante toda mi vida universitaria.

Al Honorable Gobierno Provincial de Tungurahua y en especial a los integrantes de la Dirección de Sistemas y del Programa de Aguas y Cuencas de Tungurahua que gracias a su amistad y enseñanzas he podido crecer en el ámbito profesional y como persona.

Al equipo técnico de la Cooperación Alemana GIZ – Sierra Centro por la oportunidad que me brindaron para desarrollar este proyecto, sin ellos esto no sería posible.

A mi tutor, quien con su experiencia, conocimiento y motivación me ayudo a desarrollar este proyecto.

Paul Alberto Panata Buñay

ÍNDICE DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE DE CONTENIDOS	viii
ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS.....	xi
RESUMEN EJECUTIVO	xii
ABSTRACT.....	xiii
CAPÍTULO I MARCO TEÓRICO.....	1
1.1 Antecedentes Investigativos.....	1
1.2 Contextualización del problema.....	2
1.3 Fundamentación teórica	4
1.3.1 Estación Meteorológica.....	4
1.3.2 Parámetros meteorológicos	4
1.3.4 Lenguajes de programación	6
1.3.5 Base de datos.....	7
1.3.6 Frameworks y librerías.....	7
1.3.7 Servicios y protocolos	8
1.3.8 Metodologías de desarrollo de Software.....	9
1.4 Objetivos	15
CAPÍTULO II METODOLOGÍA.....	16

2.1 Materiales.....	16
2.2 Métodos.....	16
2.2.1 Modalidad de la Investigación	16
2.2.2 Población y Muestra.....	16
2.2.3 Recolección de Información	17
2.2.4 Procesamiento y Análisis de Datos	22
2.2.5 Desarrollo del Proyecto.....	23
CAPÍTULO III RESULTADOS Y DISCUSIÓN.....	24
3.1 Desarrollo de la propuesta.....	24
3.1.1 Análisis de la situación actual	24
3.1.2 Aplicación de la metodología.....	27
3.1.3 Descripción de la base de datos	35
3.1.4 Configuración del proyecto.....	40
3.1.5 Procesamiento de datos	42
3.1.6 Detección anticipada de un evento de helada	48
3.1.7 Emisión de alertas mediante mensajes SMS.....	58
3.1.8 Publicación de datos.....	61
3.1.9 Configuración de parámetros de la emisión alertas	63
3.1.10 Ejecución de pruebas.....	64
CAPÍTULO IV CONCLUSIONES Y RECOMENDACIONES.....	68
5.1 Conclusiones	68
5.2 Recomendaciones.....	69
MATERIALES DE REFERENCIA	70
Referencias Bibliográficas	70
Anexos	73

ÍNDICE DE TABLAS

Tabla 1 Diferencias entre Scrum y Kanban	14
Tabla 2 Técnicos del HGPT y GIZ	17
Tabla 3 Ficha de observación. Oficinas del Programa de Aguas y Cuencas de Tungurahua	17
Tabla 4 Ficha de observación. Unidad Educativa de Tamboloma.....	18
Tabla 5 Matriz de tabulación entrevistas	19
Tabla 6 Descripción de las Historias de Usuario	28
Tabla 7 Descomposición de historias de usuario en tareas	29
Tabla 8 Mapeo de colecciones a entidades MongoEngine	42
Tabla 9 Controles de calidad aplicados a los parámetros meteorológicos.....	45
Tabla 10 Preguntas claves sobre los eventos de heladas en Tamboloma	52
Tabla 11 Contenido de los mensajes SMS.....	58
Tabla 12 URIs del servicio web de envío de mensajes SMS.....	59
Tabla 13 URIs del servicio web de heladas	61
Tabla 14 Casos de prueba para las funciones y métodos	65

ÍNDICE DE FIGURAS

Fig. 1 Proceso de desarrollo de Scrum.....	11
Fig. 2 Estructura del sistema de procesamiento de información hidrometeorológica de Tungurahua.....	25
Fig. 3 Infraestructura de servidores.....	26
Fig. 4 Flujo de trabajo del tablero Kanban.....	27
Fig. 5 Iteraciones basadas en Scrum	32
Fig. 6 Iteraciones dirigidas por eventos	32
Fig. 7 Tablero Kanban.....	34
Fig. 8 Modelo relacional de las tablas Grupos y Contactos.....	36
Fig. 9 Estructura de la base de datos	39
Fig. 10 Estructura de los archivos de configuración	40
Fig. 11 Ejemplo de log	42
Fig. 12 Estaciones meteorológicas utilizadas en el estudio de detección anticipada de una helada.....	48
Fig. 13 Evento de helada en Tamboloma en la madrugada del 06/09/2018.....	49
Fig. 14 Evento de helada en Tamboloma en la madrugada del 07/09/2018.....	50
Fig. 15 Evento de helada en Tamboloma en la madrugada del 19/09/2018	50
Fig. 16 Evento de helada en Tamboloma en la madrugada del 24/10/2018	51
Fig. 17 Temperatura mínima de cada 5 minutos en la estación Tamboloma desde el 09/10/2018 17:00 hasta el 24/10/2018 11:40.....	52
Fig. 18 Algoritmo de detección de heladas y emisión de alertas tempranas.....	55
Fig. 19 Proceso de detección de heladas y emisión de alertas tempranas.....	57
Fig. 20 Página de inicio del servicio de mensajería del HGPT.....	59
Fig. 21 Mensajes SMS de inicio y fin de riesgo de helada	60
Fig. 22 Interfaz web de registros históricos de eventos de heladas.	62
Fig. 23 Gráfica de múltiple con la información de temperatura, humedad relativa y punto de rocío.....	62
Fig. 24 Formulario de ajuste de parámetros.....	63
Fig. 25 Listado de suscriptores de la estación de Tamboloma.....	64
Fig. 26 Ejecución de pruebas unitarias	66
Fig. 27 Ejecución de las pruebas funcionales	67

RESUMEN EJECUTIVO

El cambio climático ha provocado que fenómenos meteorológicos como las heladas ocurran de forma inesperada, con más intensidad y en lugares donde no ha ocurrido normalmente lo cual trae consigo una serie de problemas que afectan principalmente a los agricultores debido al potencial daño que generan a los cultivos.

Gracias a los avances tecnológicos en el campo de la agricultura y meteorología han permitido la aparición de nuevas tecnologías que ayudan a advertir sobre la aparición de los eventos de heladas, sin embargo, este tipo de soluciones solo ha sido implementada por grandes empresas del sector privado, excluyendo a los pequeños agricultores que en el caso de la provincia de Tungurahua la mayoría son personas indígenas y de bajos recursos.

Con este antecedente, la investigación realizada presenta un software de control meteorológico desarrollado en Python que permitirá alertar en tiempo real mediante mensajes SMS sobre la posible aparición de un evento de helada a grupo de personas suscritas al Sistema de Alerta Temprana frente a Heladas (SATH) que por ahora se realiza el piloto en la comunidad Yatzaputzan de la provincia de Tungurahua.

La información climatológica es obtenida de una estación meteorológica con transmisión de datos en tiempo real, instalada en la comunidad de Tamboloma en la provincia de Tungurahua, cerca de la zona de intervención. Luego se valida la información por una serie de controles definidos por la Organización Meteorológica Mundial (OMM) para finalmente monitorear la temperatura, humedad relativa y punto de rocío.

Por otra parte, para el envío de alertas mediante mensajes SMS se utilizó el servicio de mensajería del Honorable Gobierno Provincial de Tungurahua (HGPT) a través del consumo de un servicio web de arquitectura REST.

Palabras Clave: Heladas, Temperatura, Python, SMS, REST, Kanban.

ABSTRACT

Climate change has caused meteorological phenomena such as frost to occur unexpectedly, more intensely and in places where it has not occurred normally, resulting in a series of problems that mainly affect farmers due to the potential damage they generate to crops.

Thanks to technological advances in the field of agriculture and meteorology have allowed the emergence of new technologies that help to warn about the appearance of frost events, however, this type of solution has only been implemented by large private sector companies, excluding small farmers that in the case of the province of Tungurahua most people are indigenous and low income.

With this background, the research presents a meteorological control software developed in Python that will alert in real time through SMS messages about the possible appearance of a frost event to a group of people subscribed to the Early Warning System against Frost (EWS) that for now is carried out by the pilot in the community Yatzaputzan in the province of Tungurahua.

The meteorological information is obtained from a weather station with real-time data transmission, installed in the community of Tamboloma in the province of Tungurahua, near the intervention area. The information is then validated by a series of controls defined by the World Meteorological Organization (WMO) to finally monitor temperature, relative humidity and dew point.

On the other hand, SMS alerts were sent using the messaging service of the Honorable Gobierno Provincial de Tungurahua (HGPT) through the consumption of a web service of REST architecture.

Keywords: Frost, Temperature, Python, PHP, SMS, REST, Kanban.

CAPÍTULO I

MARCO TEÓRICO

1.1 Antecedentes Investigativos

Sobre el tema de detección de heladas y emisión de alertas se han realizado varios proyectos de investigación de los cuales podemos destacar los siguientes:

El proyecto de investigación de Carlos Arcos con tema “Diseño e implementación de un sistema electrónico para el monitoreo, control de la helada y riego, en huertos de fresas por medio de redes de sensores inalámbricos con hardware libre” enfatiza la importancia de utilizar un sistema de supervisión meteorológico de control de helada, además recomienda el uso de una plataforma Web con el fin de que especialistas realicen investigaciones sobre el comportamiento de los parámetros meteorológicos durante un episodio de helada. Otro aspecto que recomienda es el uso sistema de aviso mediante la red GSM con la finalidad de enviar al usuario información relevante sobre el terreno. [1]

En el proyecto de investigación de Evelyn Castro con tema “Sistema de control electrónico de plagas, calefacción y regadío en las plantaciones de mora ubicadas en la Provincia de Tungurahua, Parroquia Huachi Grande Sector El Arbolito” concluye que por medio del monitoreo de la temperatura a tiempo real permite controlar el riego del terreno y la calefacción de este como efecto preventivo contra las heladas. [2]

Otro ejemplo de la automatización de detección de heladas mediante software es el proyecto de investigación realizado por Carlos Araujo con tema “Diseño e implementación de un prototipo electrónico para el monitoreo y señalización de la producción agrícola de quinua a cielo abierto en la hacienda la Cantora” donde señala que el monitoreo sobre el clima permite que la presencia de un fenómeno de helada no

represente una pérdida sustancial para el cultivo de quinua. También indica que los registros históricos y la representación gráfica sobre el comportamiento climatológico ayudan al administrador de la Hacienda la Cantora a la toma de decisiones sobre los métodos de protección frente a heladas. [3]

Del mismo modo Williams Santos y Boris Zarabia en su trabajo de investigación “Diseño, automatización y supervisión de un sistema de riego localizado de alta frecuencia, protección contra heladas y dosificación de fertilizantes para la inyección que requiere el cultivo de fragona (frutilla)” presenta la plataforma LabView 7.0 Express con la que se consiguió desarrollar y generar la aplicación para el registro de los parámetros meteorológicos y el control en la operación del sistema. [4]

1.2 Contextualización del problema

Las heladas son fenómenos meteorológicos de gran preocupación en la actividad agrícola a nivel mundial, debido a que pueden producir daños importantes en los cultivos, lo que deriva en considerables pérdidas que afectan significativamente la economía del agricultor.

Si bien las heladas son un fenómeno recurrente, en los últimos años se ha observado como la variabilidad del clima ha hecho que se produzcan en forma inesperada y en zonas donde no han ocurrido normalmente [5]. Este hecho se agrava más si se considera los resultados publicados en un estudio realizado por el Instituto Internacional de Investigación sobre Políticas Alimentarias (IFPRI, por sus siglas en inglés) donde menciona que la agricultura y el bienestar humano se verán afectados negativamente por el cambio climático, además que las poblaciones más afectadas serán de los países en vías de desarrollo [6].

Por tal motivo, gobiernos, organizaciones y empresas privadas han puesto un gran esfuerzo en adoptar soluciones que van desde el mejoramiento genético de las semillas hasta la implantación de Sistemas de Alerta Temprana (SAT). Esta última como resultado de la agricultura de precisión y muy popular dentro las empresas privadas, sin embargo, en años recientes han tomado gran interés por los gobiernos de varios países dado que permite advertir a los agricultores ubicados en varias zonas sobre la posible aparición de un evento de helada, con esta información ellos pueden accionar

algún método de protección activa, tales como estufas, ventiladores o riego por aspersión con el fin de mitigar los efectos de las temperaturas bajo cero sobre sus cultivos.

En el Ecuador este fenómeno meteorológico no ha sido tratado con la amplitud y sistematización que se requiere, a pesar de las repercusiones socioeconómicas que acarrea. Las heladas afectan principalmente a los agricultores de la región sierra; la mayoría de ellos indígenas, desde ya vulnerables debido a la discriminación, pobreza, zona geográfica donde se asientan y sobre todo que la mayoría de ellos depende de la agricultura para subsistir.

Para los agricultores de la provincia de Tungurahua las heladas son un factor de alto riesgo, pues el impacto de estas genera pérdidas de hasta el 100% de sus cultivos. Tal es el caso de Patricia Guevara, agricultora del sector de La Florida, cantón Cevallos; quien en junio de 2018 sus cultivos de alfalfa fueron afectados por las heladas, lo que generó que su producción bajara al menos el 50% [7]. Otro caso es el de María Elena Punina agricultora de la comunidad de Yatzaputzan, Parroquia Pillahuin que en el segundo semestre de 2018 sus cultivos de papas fueron fuertemente afectadas, lo que desencadenó que la producción bajara drásticamente.

Por otra parte, en 2018 el Honorable Gobierno Provincial de Tungurahua (HGPT) con asesoría de El Programa Cambio Climático, Biodiversidad y Desarrollo Sostenible (ProCamBio II) de la Cooperación Alemana al Desarrollo – GIZ iniciaron un proyecto piloto con nombre “Sistema de Alerta Temprana frente Heladas en la comunidad Yatzaputzan” con la finalidad de combatir los daños producidos por las heladas en los cultivos. Para dicho proyecto se utilizará el sistema de riego colectivo tecnificado y la red de estaciones hidrometeorológicas, ambas instaladas y mantenidas por el HGPT. En el caso de las estaciones toda la información hidrometeorológica es procesada, validada y publicada en un sistema web, sin embargo, actualmente este sistema no cuenta con la funcionalidad de detección heladas por lo que surge la problemática y necesidad de desarrollar uno para continuar con el proyecto.

1.3 Fundamentación teórica

1.3.1 Estación Meteorológica

Una estación meteorológica es un lugar donde se realizan observaciones y mediciones sobre varios parámetros meteorológicos, utilizando instrumentos apropiados, con el fin de establecer el comportamiento atmosférico de las zonas de un territorio [8].

Las estaciones meteorológicas se clasifican de acuerdo con las normas técnicas de la OMM.

Según para los fines que fueron instaladas, tenemos:

- Pluviométricas
- Climatológicas principales y ordinarias
- Agrometeorológicas
- Sinópticas principales y secundarias
- Meteorológicas especiales

1.3.2 Parámetros meteorológicos

Temperatura

Representa el estado termodinámico de un cuerpo y su valor está determinado por la dirección del flujo neto de calor entre dos cuerpos.

Para fines meteorológicos se miden las temperaturas de numerosos medios; en nuestro caso utilizaremos la temperatura del aire, la cual la OMM la define como “la temperatura leída en un termómetro expuesto al aire, protegido de la radiación solar directa”. [9]

Humedad Relativa

La OMM define la humedad relativa como la razón, expresada en porcentaje, entre la presión de vapor observada y la tensión del vapor saturante con respecto al agua a la misma temperatura y presión. [9]

Precipitación

La OMM define la precipitación como el producto líquido o sólido de la condensación del vapor de agua que cae de las nubes o del aire y se deposita en el suelo. Dicho término comprende la lluvia, el granizo, la nieve, el rocío, la cencellada blanca, la escarcha y la precipitación de la niebla. La unidad de medida habitual de la precipitación es en milímetros (mm). [9]

Punto de rocío

El punto de rocío es la temperatura a la que el aire debe enfriarse para que la humedad contenida en este se condense. La temperatura de rocío es el resultado de función matemática en la que interviene la temperatura y la humedad relativa.

1.3.3 Fenómeno meteorológico de helada

Existen varias definiciones para la palabra helada, pero la mayoría de los autores la definen como la ocurrencia de una temperatura del aire de 0 °C o inferior, no obstante, la palabra es ampliamente utilizada por el público, principalmente por los agricultores para describir un fenómeno meteorológico cuando los cultivos y otras plantas experimentan daño por congelación. [10]

Tipos de heladas

Por su origen climatológico, los eventos de heladas se clasifican en:

- Heladas por advección
- Heladas por radiación.

Las heladas por advección se producen cuando el aire frío fluye en un área para reemplazar el aire más caliente que estaba presente antes del cambio meteorológico. Está asociada con condiciones de nubosidad, vientos de moderados a fuertes, sin inversión de temperatura y baja humedad. A menudo las temperaturas serán inferiores a 0 °C durante todo el día.

En cambio, las heladas por radiación se producen por la pérdida calor que sufren las plantas y suelo durante la noche por el proceso de radiación. Este tipo de helada se caracteriza por un cielo despejado, en calma o con poco viento, inversión de temperatura, temperaturas del punto de rocío bajas y temperaturas del aire que normalmente caen por debajo de 0 °C durante la noche pero que están por encima de 0 °C durante el día. [11]

Por su aspecto visual, los eventos de heladas se clasifican en:

- Helada blanca
- Helada negra

La helada blanca se produce cuando la temperatura del aire alcanza el punto de rocío, de modo que existe condensación lo que origina la escarcha.

En cambio, la helada negra se origina cuando el aire tiene poco vapor de agua (humedad baja) y la temperatura del punto de rocío es inferior a 0° C; de modo que existe escasa condensación y nula formación de hielo sobre la planta lo que genera quemaduras en los tejidos de las plantas. Este tipo de helada provoca daños más severos que las heladas blancas. [11]

1.3.4 Lenguajes de programación

Python

Es un lenguaje de programación interpretado y multiparadigma que permite crear código de forma rápida y con una sintaxis fácil de leer. Estas características le han permitido a Python gozar de una gran popularidad y usos que van desde el scripting, procesamiento de datos, inteligencia artificial e incluso en el desarrollo Web con frameworks como Django o Flask.

Todos estos usos nunca hubiesen sido posible sin las contribuciones de la comunidad que mantiene a Python ya cada día potencian el lenguaje con más características, librerías o frameworks que son utilizados por personas que inician en el mundo de la programación o hasta grandes empresas como Google, Facebook, la NASA, Dropbox entre otros. [12]

PHP (Hypertext Preprocessor)

Es un lenguaje de programación de scripting de propósito general, comúnmente utilizado para el desarrollo web. [13]

1.3.5 Base de datos

MongoDB

Es una base de datos NoSQL basada en documentos que destaca por su rapidez, flexibilidad y la alta disponibilidad. [14]

1.3.6 Frameworks y librerías

CodeIgniter

Es un framework para la creación de sitios web con PHP enfocado en el desarrollo rápido. [15] Entre las características principales de CodeIgniter se destacan las siguientes:

- Es muy liviano, en su versión 3 ocupa tan sólo 2MB, incluyendo la guía del usuario.
- Se puede utilizar el patrón de arquitectura de software MVC (Modelo – Vista – Controlador)
- Cuenta con protección contra ataques CSRF y XSS.
- La instalación es muy sencilla.

Highcharts

Es una librería para la creación de gráficas estadísticas en proyectos web y móvil. [16] Ofrece una amplia variedad de gráficas que trabajan con datos de tipo CSV, JSON o datos cargados por defecto y permite exportar las gráficas a formatos como PNG, JPG, PDF o SVG.

MongoEngine

Es un Document-Object Mapper (algo así como ORM para bases de datos no relaciones) para trabajar con MongoDB desde Python. [17]

pytest

Es un framework para realizar pruebas en Python en varios niveles de prueba de software [18]. La facilidad para escribir casos de prueba incluso de casos complejos y la legibilidad en el código ha hecho de pytest la herramienta principal para pruebas de varios proyectos en el mundo incluido de empresas como Dropbox y Mozilla.

1.3.7 Servicios y protocolos

FTP (File Transfer Protocol)

Es un protocolo de transferencia de archivos para redes TCP/IP basado en la arquitectura cliente-servidor [19]. FTP realiza la transferencia de archivos de forma sencilla, ocultando todos los detalles de la implementación de cómo se mueven los archivos de un lugar a otro.

Para utilizar el protocolo se necesita de un cliente y servidor FTP, donde el cliente se conecta al servidor para descargar, subir, mover o borrar archivos independientemente del sistema operativo y con un nivel alto de confiabilidad de que no haya pérdida de datos dado que FTP utiliza TCP en la capa de transporte.

Servicios Web de REST

Es un servicio web que implementa un conjunto de principios arquitectónicos de REST que se centran en los recursos de un sistema y la transferencia a través del protocolo HTTP. [20]

El diseño de servicios web de REST siguen estos cuatro principios:

1. Utilizar métodos HTTP de forma explícita.
2. Ser sin estado.
3. Exponer las URIs como estructura de directorios.
4. Intercambiar información con XML, JSON, o ambos.

GPRS (General Packet Radio Service)

Es un servicio de comunicación inalámbrica de paquetes que permite la transmisión de datos y la conexión permanente a Internet a teléfonos móviles y usuarios de ordenador. El servicio GPRS se comunica a través del Sistema global para las comunicaciones móviles (GSM). [21]

SMS (Short Message Service)

Es un servicio de mensajería disponible en la red GSM para enviar y recibir mensajes de texto de hasta 160 caracteres a teléfonos móviles.

1.3.8 Metodologías de desarrollo de Software

En la industria de software existen un gran número de metodologías de desarrollo de software, cada una con sus ventajas, desventajas y casos de aplicación. Antes de que apareciera el término “ágil” los equipos de desarrollo utilizaban metodologías que seguían una serie de procedimientos y pasos rígidos. [22]

Comúnmente los proyectos de software que utilizan este tipo de metodologías tienden a fracasar por no cumplir en el tiempo establecido, sobrepasar el presupuesto o que el software final no cumple con los requisitos que el cliente pidió.

Por tal motivo en febrero de 2001, un grupo de 17 expertos de la industria del software decidieron reunirse en Snowbird, Utah con el fin de tratar nuevos enfoques de desarrollo de software que permitan a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Tras esta reunión aparece el término ágil y consigo el “manifiesto ágil”, una serie de cuatro valores en los que se basa el desarrollo ágil de software. Además, se creó *The Agile*

Alliance, una organización sin ánimo de lucro dedicada a promover los conceptos relacionados con el desarrollo ágil de software. [23]

El manifiesto ágil valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.
- Software que funcione en lugar de conseguir una buena documentación
- La colaboración con el cliente más que la negociación de un contrato.
- Responder a los cambios más que seguir estrictamente un plan.

Metodologías ágiles

Las metodologías ágiles son técnicas de desarrollo de software que permiten adaptar la forma de trabajo a las necesidades del proyecto, consiguiendo flexibilidad y capacidad de realizar cambios de una manera más rápida.

Estas metodologías se caracterizan por un desarrollo iterativo e incremental, promoviendo el trabajo en equipo, retroalimentación y sobre todo la comunicación con el cliente al punto de requerir un representante de él durante el desarrollo. [24]

Metodología ágiles es un término que cubre varias metodologías de desarrollo como son Scrum, Extreme Programming (XP), Kanban, entre otras.

Scrum

Scrum es una de las metodologías ágiles más utilizada por la industria del software. Se presta a proyectos de cualquier tamaño y complejidad. Esta metodología se centra en la colaboración en equipo, la participación del cliente y la entrega continua.

El marco técnico de scrum está formado por:

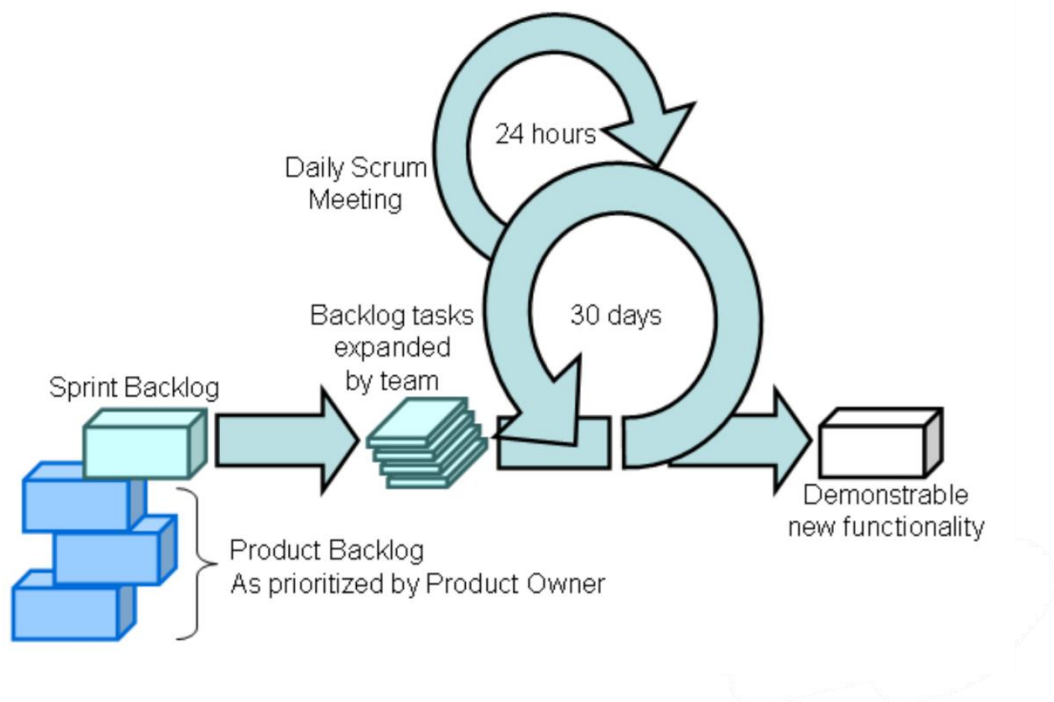
- Roles
 - El equipo scrum (Scrum Team)
 - El dueño del producto (Product Owner)
 - El Scrum Master

- Artefactos
 - Pila del producto (Product Backlog)
 - Pila del sprint
 - Incremento
- Eventos
 - Sprint
 - Reunión de planificación del sprint (Sprint Planning Meeting)
 - Scrum diario
 - Revisión del sprint
 - Retrospectiva del sprint

Un sprint es un ciclo o iteración de trabajo que dura máximo un mes en el que se produce una parte del producto terminada y funcionalmente operativa (incremento). [25]

El proceso de desarrollo de Scrum es el que se muestra en la figura 1.

Fig. 1 Proceso de desarrollo de Scrum



Fuente: Agile Software Development with Scrum by Ken Schwaber and Mike Beedle

Todo comienza en el **Product Backlog**, que es una lista de los requerimientos del producto definidos por los miembros del proyecto.

Los proyectos se realizan durante una serie de iteraciones llamadas **Sprints**. Al comienzo de cada Sprint tiene lugar una **Sprint Planning Meeting** durante la cual el **Product Owner** prioriza el Product Backlog y el **Scrum Team** selecciona las tareas que serán completadas durante el Sprint que va a comenzar. Al finalizar las tareas son removidas del Product Backlog para ser llevadas al **Sprint Backlog**.

Durante el Sprint el equipo se mantiene en contacto a través de las **Daily Meetings**. Y al final del Sprint debe mostrar la funcionalidad completa en la **Sprint Review Meeting**. [26]

Kanban

El término Kanban hace referencia a un método creado por Toyota para controlar el avance del trabajo, en el contexto de una línea de producción. Debido a su éxito, Kanban fue adoptado por varias industrias, en la que el software no sería la excepción.

David J. Anderson, impulsor de Kanban, formuló el método Kanban como una aproximación al proceso evolutivo e incremental y al cambio de sistemas para las organizaciones de trabajo. [27]

El método Kanban es un conjunto de prácticas y principios que permiten mejorar el proceso de desarrollo de software.

El método considera tres principios básicos:

1. Empezar con lo que se está haciendo
2. Decidir que se va a seguir cambios incrementales y evolucionarios
3. Respetar el proceso en curso, sus roles, responsabilidades y títulos

Además, contempla seis practicas:

1. Visualizar el flujo de trabajo

Aquí aparece el tablero Kanban, con esta herramienta se podrá ver rápidamente el flujo real del proyecto. El tablero deberá estar situado en un lugar visible para todos los miembros involucrados en el proyecto. También se puede

utilizar tarjetas de colores para identificar a quien pertenece, el tipo de tarea (funcionalidades, bugs ...) o la gravedad de la tarea.

2. Limitar el trabajo en curso

Permite a los equipos de desarrollo establecer un límite al número de elementos de trabajo activos en cada fase del flujo de trabajo con la finalidad de que puedan trabajar a un ritmo sostenible. El número de elementos se conoce como WIP (Work In Progress).

3. Dirigir y gestionar el flujo

En el Método Kanban se percibe el trabajo como un flujo de valor (Lean flow). Un buen flujo proviene cuando se finaliza las tareas sin sobrecargar el sistema, en cambio un mal flujo proviene cuando se presentan cuellos de botellas o se pierde el tiempo realizando tareas que no aportan valor.

Para mejorar el flujo y prevenir la aparición de cuellos de botella se recomienda utilizar métricas e inspeccionar el flujo de trabajo.

4. Hacer las Políticas de Proceso Explícitas

Al comenzar un proceso han de quedar definidos ciertos conceptos básicos que faciliten el flujo de las tareas a través del tablero Kanban. Todas las políticas deben estar claramente escritas e incluidas en el tablero. Además, deben ser respetados por todos los miembros del equipo y si representan un problema más que una solución o bien si pueden ser mejorado podrán ser discutidos en cualquier momento por el equipo. [28]

5. Circuitos de retroalimentación

Los circuitos de retroalimentación permiten comprobar cómo evoluciona el proceso de desarrollo. Para lograr esto a menudo se realiza reuniones, como en Scrum, pero con la diferencia de que se deberá tener el tablero Kanban cerca.

6. Utilizar modelos para reconocer oportunidades de mejora

La gran idea detrás del método Kanban aplicado al trabajo del conocimiento es mejorar evolutivamente desde el estado actual usando pequeños pasos. [29]

En resumen, Kanban y Scrum son herramientas de proceso que ayudan a trabajar más eficazmente. Como cualquier herramienta, no son ni perfectas ni completas. No dicen todo lo que se debe hacer, solo proporcionan ciertas restricciones y directrices. Por ejemplo, Scrum obliga a tener iteraciones de duración fija y equipos interdisciplinarios, y Kanban obliga a usar tableros visibles y a limitar el tamaño de tus colas.

Aunque Kanban y Scrum pertenecen al grupo de metodologías ágiles, presentan una serie de diferencias como se observa en la tabla 1.

Tabla 1 Diferencias entre Scrum y Kanban

Scrum	Kanban
Las iteraciones deben ser de tiempo fijo	No se prescriben iteraciones de tiempo fijo.
La velocidad es la métrica principal para la planificación y la mejora del proceso.	Lead Time (tiempo de entrega o tiempo medio que tarda una petición en salir del ciclo), duración del ciclo y trabajo en curso con las principales métricas para la planificación y la mejora del proceso.
Las funcionalidades deben dividirse en partes que puedan completarse en un sprint.	No hay ninguna prescripción en cuanto al tamaño de las divisiones.
Se debe definir roles: Propietario del producto, experto en scrum, equipo de desarrollo	No se obliga definir roles.
Al finalizar un sprint se limpia el tablero.	Se mantienen las tareas al finalizar el flujo.
Se deben realizar estimaciones.	Las estimaciones son opcionales.
No se pueden añadir tareas en medio de una iteración.	Siempre que haya capacidad disponible, se pueden añadir tareas.

Elaborado por: El investigador

1.4 Objetivos

A modo general se plantea implementar un software de control meteorológico para la detección de heladas y emisión de alertas tempranas en el Geoportal del Honorable Gobierno Provincial de Tungurahua.

A partir de este objetivo general, se establecen los siguientes objetivos específicos:

- Estudiar la infraestructura informática para el control, validación y publicación de datos de la red hidrometeorológica de Tungurahua.
- Identificar los patrones meteorológicos antes y durante un episodio de helada.
- Validar la información climatológica según los modelos y normas definidas por la Organización Meteorológica Mundial.
- Implementar el proceso de detección de heladas y emisión de alertas tempranas.

CAPÍTULO II METODOLOGÍA

2.1 Materiales

Los materiales utilizados para la recolección de la información fueron: fichas de observación y formularios para entrevistas.

2.2 Métodos

2.2.1 Modalidad de la Investigación

La investigación se realizó bajo las siguientes modalidades:

- Investigación bibliográfica – documental, porque se tomará información de Internet, libros virtuales, tesis de grados, videos, informes, proyectos, revistas e informes, la cual servirá como sustento investigativo del proyecto.
- Investigación de campo, dado que fue necesario realizar un estudio del sitio donde se genera el problema.
- Investigación experimental, en vista que se llevará a cabo pruebas de funcionamiento de la emisión de alertas mediante mensajes SMS.

2.2.2 Población y Muestra

La presente investigación por su característica no amerita definir una muestra puesto que solo se trabajará con el equipo técnico del HGPT y de la GIZ.

En la tabla 2 se muestra el número de técnicos que colaboraron en el desarrollo de la presente investigación.

Tabla 2 Técnicos del HGPT y GIZ

Institución	Dirección / Programa	Cantidad	Porcentaje
HGPT	Programa de Aguas y Cuencas de Tungurahua (PACT)	1	0.25 %
	Dirección de Sistemas	1	0.25 %
GIZ	Programa Cambio Climático, Biodiversidad y Desarrollo Sostenible (ProCamBio II)	2	0.50 %
Total		4	100 %

Elaborado por: El investigador

2.2.3 Recolección de Información

Para el trabajo de investigación se realizó la recolección de información en base a la observación de campo y entrevistas.

La observación se llevó a cabo en dos sitios. El primero en las oficinas del Programa de Aguas y Cuencas de Tungurahua perteneciente a la Dirección de Recursos Hídricos y Conservación Ambiental del HGPT, en este lugar se obtuvo información acerca de la red hidrometeorológica de la provincia de Tungurahua. Los temas que se abordó se muestran en la ficha de observación de la tabla 3.

Tabla 3 Ficha de observación. Oficinas del Programa de Aguas y Cuencas de Tungurahua

FICHA DE OBSERVACIÓN	
Número	1
Lugar	Oficinas del Programa de Aguas y Cuencas de Tungurahua
Fecha	06/11/2018
Elaborado por	Paul Panata
Categoría	Lo observado
1. Estaciones hidrometeorológicas	La ubicación de las estaciones. Tipo de estaciones. Número de estaciones pertenecientes al HGPT y de convenios. Sensores de medición.

2. Datos climatológicos	Parámetros meteorológicos. Procesamiento de la información climatológica. Uso de la información climatológica. Recolección de información.
3. El Geoportal y el sistema web para procesamiento de datos hidrometeorológicos.	Línea de tiempo del desarrollo del Geoportal. Funcionalidades y fallos del sistema web para procesamiento de datos hidrometeorológicos. Técnicos que mantienen el Geoportal. Personas que utilizan el Geoportal.

Elaborado por: El investigador

El segundo sitio fue en la Unidad Educativa de Tamboloma, zona donde está instalada la estación meteorológica para la detección anticipada de eventos de heladas, en el lugar se obtuvo información de los aparatos de medición del clima y transmisión de datos. Los temas que se abordó se muestran en la ficha de observación de la tabla 4.

Tabla 4 Ficha de observación. Unidad Educativa de Tamboloma

FICHA DE OBSERVACIÓN	
Número	2
Lugar	Unidad Educativa de Tamboloma
Fecha	24/11/2018
Elaborado por	Paul Panata
Categoría	Lo observado
1. Estación meteorológica	Zona de instalación. Sensores de medición del clima. Configuración del datalogger.
2. Trasmisión de datos	Conexión utilizada para el envío de datos hacía el servidor FTP. Intervalos de transmisión.
2. Zona de intervención	Terrenos de los agricultores. Tipo de cultivos. Sistema de riego parcelario.

Elaborado por: El investigador

Las entrevistas fueron realizadas a los técnicos de la GIZ y HGPT. Con el técnico de la Dirección de Sistemas del HGPT se trató temas de la infraestructura informática, en cambio para temas sobre las estaciones meteorológicas, tratamiento de la información climatológica y el sistema web de control, validación y publicación de datos meteorológicos se entrevistó al técnico que administra el Geoportal, perteneciente al Programa de Aguas y Cuencas de Tungurahua del HGPT. Por último, se entrevistó a dos técnicos de la GIZ para conocer el proceso de detección anticipada de una helada, el impacto de las heladas y formas como advertir a las personas sobre estos eventos extremos. En la tabla 5 se presenta la matriz de entrevista realizada a los técnicos.

Tabla 5 Matriz de tabulación entrevistas

Entrevista a los técnicos del HGPT y GIZ		
Lugar	Oficina de la GIZ	
Fecha	18/10/2018	
Entrevistados	Ing. David Mantilla (HGPT), Ing. Mario Torres (HGPT), M.Sc. Patrick Reuter (GIZ) y M.Sc. Susana Perez de Mora (GIZ)	
Pregunta	Técnico	Respuesta
1. ¿Cuáles son las estaciones meteorológicas con transmisión de datos en tiempo real?	Ing. David Mantilla (HGPT)	Las estaciones de Mula Corral, Chiquihurco, Baños Parque de la Familia y Tamboloma. La estación meteorológica de Tamboloma fue la última en instalarse.
2. ¿Qué tipo de datalogger utilizan las estaciones meteorológicas con transmisión de datos en tiempo real?	Ing. David Mantilla (HGPT)	Un datalogger Sutron 8310 excepto la estación de Tamboloma.
3. ¿Cuál es el tipo de conexión que utilizan las estaciones meteorológicas con transmisión de datos en tiempo real?	Ing. David Mantilla (HGPT)	Las estaciones de Mula Corral y Chiquihurco están integradas a una red SCADA, por otro lado, la estación de Baños utiliza un enlace web y la estación de

		Tamboloma utiliza una conexión GPRS con la operadora CLARO.
4. ¿Cuál es el intervalo de registro de las estaciones meteorológicas con transmisión de datos en tiempo real?	Ing. David Mantilla (HGPT)	Las estaciones de Mula Corral, Chiquihurco y Baños Parque de la Familia registran los datos cada un minuto, cinco minutos y una hora. En cambio, la estación de Tamboloma lo realiza solamente cada cinco minutos.
5. ¿Cuál es el intervalo de transmisión de datos de las estaciones automáticas?	Ing. David Mantilla (HGPT)	Las estaciones de Mula Corral, Chiquihurco y Baños Parque de la Familia transmiten los datos cada hora. En cambio, la estación de Tamboloma lo realiza solamente cada cinco minutos.
6. ¿Cuáles son los controles de calidad que validan la información climatológica?	Ing. David Mantilla (HGPT)	Los controles de calidad son definidos por la Organización Meteorológica Mundial y son los siguientes: <ul style="list-style-type: none"> - Control de umbrales - Control de consistencia temporal - Control de consistencia interna Los controles son para datos instantáneos de un minuto.

7. ¿El sistema web tiene la funcionalidad de monitorear el clima en tiempo real?	Ing. David Mantilla (HGPT)	No, solo son almacenados y publicados en el sitio web.
8. ¿Cuál es el propósito principal que motiva a los usuarios descargar la información climatológica?	Ing. David Mantilla (HGPT)	Para investigación
9. ¿El sistema web cuenta con la funcionalidad de detección anticipada de un fenómeno de helada ?	Ing. David Mantilla (HGPT)	Por el momento no, es lo que se busca en un inicio para la estación meteorológica de Tamboloma.
10. ¿Cuáles son las tecnologías que utilizan el sistema de información?	Ing. Mario Torres (HGPT)	El sistema web fue desarrollado con CodeIgniter, utilizó MongoDB para centralizar la información, BASH para la descarga de datos y Beanstalk para el servicio de colas.
11. ¿En qué servidor se ejecuta el sistema de información?	Ing. Mario Torres (HGPT)	En servidores CentOS versión 6.
12. ¿Cómo es el servicio de mensajería del HGPT?	Ing. Mario Torres (HGPT)	El HGPT cuenta con dos maneras de comunicación, por correo electrónico y mensajes SMS. Los mensajes SMS son contratados a la empresa Eclipsoft S.A. Existe un sitio web para enviar los mensajes y administrar los contactos. Además, cuenta con un servicio web de arquitectura REST para el uso de terceros.

13. ¿Cómo se va a determinar los parámetros de detección anticipada de una helada?	M.Sc. Patrick Reuter (GIZ)	Con un estudio de los eventos de heladas reportados por las personas de Tamboloma y Yatzaputzán.
14. ¿Cuáles van a ser las estaciones las estaciones que se utilizarán el estudio?	M.Sc. Patrick Reuter (GIZ)	La estación principal es la Tamboloma. Además, se va a apoyar en las estaciones de Mula Corral, Chiquiurco y Calamaca.
15. ¿Quiénes van a recibir los mensajes de la alerta?	M.Sc. Susana Perez de Mora (GIZ)	Agricultores de la zona de Yatzaputzán, investigadores e interesados de otras instituciones.
16. ¿Cuál es el método que utilizan los agricultores para hacer frente a los eventos de helada?	M.Sc. Susana Perez de Mora (GIZ)	Riego por aspersión

Elaborado por: El investigador

2.2.4 Procesamiento y Análisis de Datos

Después de haber analizado las entrevistas y la información recogida por la observación de campo se concluye lo siguiente:

- La información hidrometereológica de la provincia de Tungurahua registrada por la red de estaciones hidrometereológicas es procesada, validada y publicada en un sistema web.
- El HGPT cuenta con 4 estaciones meteorológicas con transmisión de datos en tiempo real, estas son Mula Corral, Chiquiurco, Baños Parque de la Familia y Tamboloma. La estación meteorológica de Tamboloma fue la última en instalarse.
- El sistema web fue desarrollado en su totalidad con herramientas de código abierto.

- Actualmente el sistema web no cuenta con la funcionalidad de detección anticipada de fenómenos de heladas.
- Se llevo a cabo un estudio sobre los eventos de heladas generados en la zona de Tamboloma y Yatzaputzán.
- La funcionalidad de detección de heladas y emisión de alertas se realizará solo en estaciones meteorológicas con transmisión de datos en tiempo real.
- Para hacer frente a las heladas los agricultores van a utilizar el sistema de riego por aspersión, instalada por el HGPT.

2.2.5 Desarrollo del Proyecto

Para el desarrollo del proyecto de se realizarán las siguientes actividades:

- Determinar las colecciones que utiliza el sistema web para almacenar la información hidrometeorológica.
- Identificar los servicios y protocolos que utiliza el sistema web.
- Observar y analizar el comportamiento del clima durante los episodios de helada.
- Automatizar la descarga de datos desde el servidor FTP.
- Definir los parámetros de detección anticipada de un evento de helada.
- Aplicar los controles de calidad a los datos dependiendo del tipo de parámetro meteorológico.
- Crear un cliente REST para el servicio de mensajería del HGPT con la finalidad de enviar las alertas mediante mensajes SMS.
- Crear una interfaz gráfica web para mostrar el historial de eventos de heladas.
- Crear una interfaz gráfica web para cambiar los parámetros de la detección anticipada de una helada y los suscriptores que recibirán la alerta.
- Realizar pruebas de funcionamiento.
- Documentar el código con la herramienta Sphinx.

CAPÍTULO III

RESULTADOS Y DISCUSIÓN

3.1 Desarrollo de la propuesta

3.1.1 Análisis de la situación actual

El Honorable Gobierno Provincial de Tungurahua, a través de la Dirección de Recursos Hídricos tiene en funcionamiento desde el 2018 un sistema web para procesar, validar y publicar los datos meteorológicos e hidrométricos registrados por la red de estaciones hidrometeorológicas de la provincia de Tungurahua.

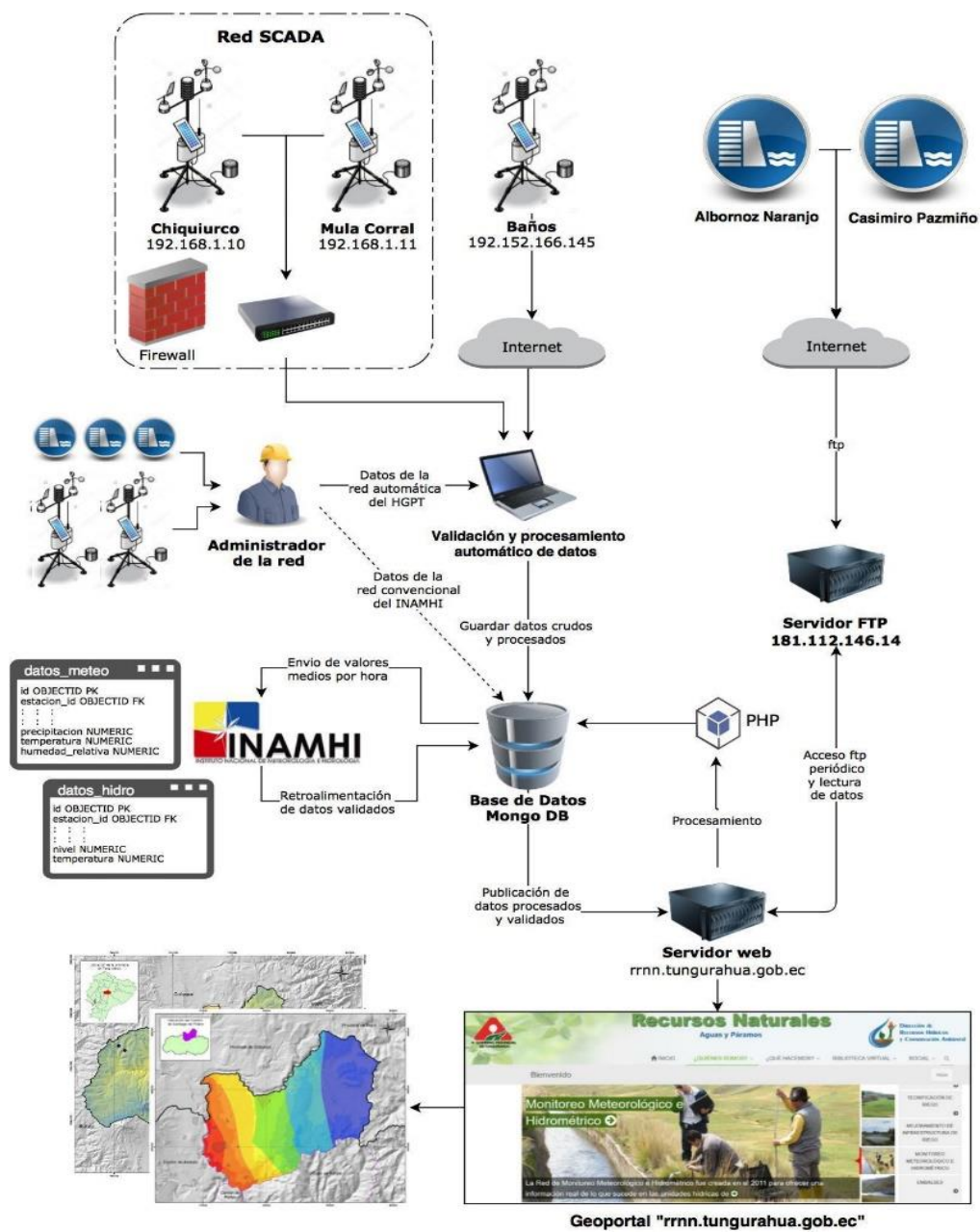
El sistema web fue desarrollado con el framework PHP CodeIgniter, utilizando MongoDB como base de datos y Beanstalk para el manejo de colas de mensajes. Por otro lado, el sistema web forma parte de una infraestructura informática que se muestra en la figura 2 y actualmente proporciona las siguientes funcionalidades:

- Almacenar en MongoDB los datos meteorológicos e hidrométricos sin validar de las siguientes fuentes: la primera fuente de información proviene de los archivos .CSV obtenidos manualmente por el administrador de la red, mientras que la segunda proviene automáticamente de las estaciones con transmisión de datos en tiempo real.
- Validar los datos meteorológicos e hidrométricos mediante un control de calidad basado en los modelos y normas que indica la OMM.
- Reducir la información climatológica en forma de promedios de hora y diarios.
- Publicar en el sitio web la información provincial de los siguientes parámetros: precipitación, temperatura, humedad, presión atmosférica, viento y el nivel de agua en canales, acequias, quebradas y ríos, a través de la hidrometría.

Con respecto a las estaciones, hasta la fecha la red hidrometeorológica de Tungurahua cuenta con 20 estaciones meteorológicas y 20 estaciones hidrométricas, de las cuales

6 son estaciones automáticas con transmisión de datos en tiempo real. De este tipo de estaciones, la última en integrarse fue la estación meteorológica de Tamboloma sin embargo el sistema web no puede procesar los datos enviados debido a que presenta un formato de datos diferente al utilizado por las demás estaciones, además que el intervalo de registro de la información climatológica por parte de la estación es cada 5 minutos en contraste a 1 minuto empleado por las demás estaciones.

Fig. 2 Estructura del sistema de procesamiento de información hidrometeorológica de Tungurahua



Elaborado por: Jaime Santana en colaboración con Patrick Reuter [30]

En el apartado de servidores, la infraestructura informática cuenta con cuatro servidores. Las características técnicas de estos son similares con la diferencia de los servicios instalados y el espacio en disco asignado.

En la figura 3 se muestra los servicios que ejecutan los servidores. Por otra parte, las funciones que realizan son las siguientes:

Servidor web

- Aloja los archivos de la aplicación web y ejecuta el servicio de colas.

Servidor de base de datos:

- Ejecuta el motor de base de datos MongoDB.

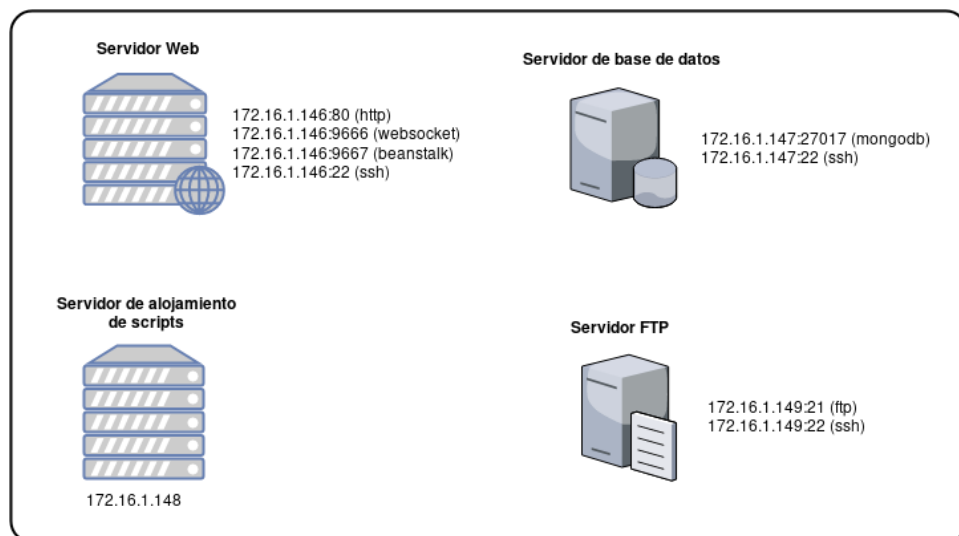
Servidor FTP

- Almacena los archivos CSV enviados por las estaciones meteorológicas con transmisión de datos en tiempo real.

Servidor de script

- Contiene los scripts BASH para descargar la información hidrometeorológica en formato CSV de las estaciones con transmisión de datos en tiempo real mediante el protocolo Telnet para luego subir estos archivos al servidor FTP.

Fig. 3 Infraestructura de servidores



Elaborado por: El investigador

3.1.2 Aplicación de la metodología

Para el proyecto se decidió utilizar Kanban debido a que otorga mayor flexibilidad en términos de planificación y ejecución en comparación con Scrum.

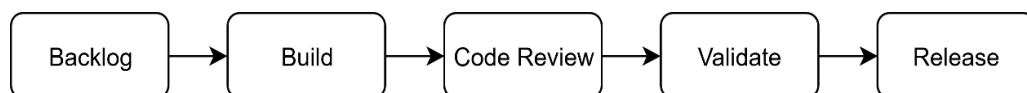
Flujo de trabajo

El primer paso para trabajar con Kanban es definir el flujo de trabajo. Para realizar esta tarea se desglosó en sus etapas de desarrollo la entrega de una característica mínima o lo que se conoce como MMF (Minimum Marketable Feature).

Un MMF es un conjunto pequeño de funcionalidades mínimas que aportan valor al usuario y por lo regular su proceso de desarrollo sigue las siguientes etapas: obtención de requerimientos, estimación, planificación, desarrollo, pruebas, validación y lanzamiento.

De las etapas antes mencionadas se procedió a mapearlas al flujo de trabajo que se presenta en la figura 4.

Fig. 4 Flujo de trabajo del tablero Kanban



Elaborado por: El Investigador

El flujo de trabajo comienza en el **Backlog**, aquí se irán añadiendo las tareas que describen un MMF o el arreglo de un bug. Luego cada tarea pasará al estado de **Build**, que significa que está en desarrollo un componente del software.

Una vez terminado el desarrollo, se continuará con el estado de **Code Review** donde se realizará una revisión del código con el apoyo de un técnico de la Dirección de Sistemas del HGPT con el objetivo de mejorar la calidad del código. Después seguirá el estado de **Validate**, en este punto se ejecutarán las pruebas unitarias y funcionales, además las de aceptación en conjunto con los técnicos del HGPT y GIZ. Finalmente,

solo si la ejecución de las pruebas fue satisfactoria se lanzará un MMF a producción, lo que significa que la tarea pasará al estado de **Release**.

Historias de usuario

Las historias de usuario son un componente clave dentro de las metodologías ágiles, y en Kanban no es la excepción. La finalidad del uso de esta herramienta es conocer las necesidades que tiene el cliente con el software. En el caso del proyecto las historias de usuario fueron definidas en conjunto con el investigador, un técnico de la Dirección de Sistemas, un técnico de la Dirección de Recursos Hídricos y Conservación Ambiental, ambos del HGPT y dos técnicos de la GIZ.

Entre los temas que se abordaron durante la reunión cabe destacar los siguientes:

- El sistema web para el procesamiento, validación y publicación de los datos hidrometeorológicos.
- La transmisión de datos de las estaciones meteorológicas automáticas.
- La información climatológica.
- Los procesos de calidad para datos meteorológicos instantáneos.
- El estudio para la detección anticipada de un evento de helada.
- Mecanismos de emisión de alertas tempranas.
- Servicio de mensajería del HGPT.

Una vez conocidos los requerimientos del cliente se procedió a crear las historias de usuario que se presentan en la tabla 6 y en más detalle en el Anexo 1.

Tabla 6 Descripción de las Historias de Usuario

ID	Como	Quiero... /Quiero que	De modo que... / Para
HU01	Administrador	El software descargue, valide y guarde los datos meteorológicos.	Investigación. Control de Calidad.
HU02	Administrador	El software monitoree en tiempo real los datos	Investigación.

		meteorológicos con la finalidad de anticipar la aparición de un evento de helada.	Emitir alertas mediante mensajes SMS.
HU03	Suscriptor al SATH	Notifique mediante un mensaje SMS el posible inicio o fin de un evento de helada.	Investigación. Proteger cultivos y animales.
HU04	Administrador Usuario	Visualizar los registros de eventos de helada.	Investigación.
HU05	Administrador	Configurar los parámetros del detector de heladas.	Reducción de falsos positivos.
HU06	Administrador	Agregar o quitar grupos suscritos al SATH	Administrar los suscriptores al SATH.
HU07	Administrador	Cambiar el intervalo de registro de datos para las estaciones meteorológicas.	Alternar el origen de datos.

Elaborado por: El Investigador

Descomposición de las historias de usuario

Descomponer las historias de usuario en tareas es una técnica que permite identificar y priorizar el trabajo. Lamentablemente realizar este procedimiento no es tan sencillo debido a que depende de muchos factores tales como el tamaño de las historias de usuario, la complejidad del proyecto, experiencia del equipo, etc. Pese a ello y partiendo de la experiencia se mapearon las historias de usuario a las tareas que se muestran en la tabla 7.

Tabla 7 Descomposición de historias de usuario en tareas

Historia de Usuario	Tareas		
HU01	<table border="1"> <tr> <td>T01</td> <td>Crear una tarea automatizada BASH para obtener los datos meteorológicos.</td> </tr> </table>	T01	Crear una tarea automatizada BASH para obtener los datos meteorológicos.
T01	Crear una tarea automatizada BASH para obtener los datos meteorológicos.		

	<table border="1"> <tr> <td>T02</td> <td>Validar los datos meteorológicos con los controles de calidad establecidos por la OMM.</td> </tr> <tr> <td>T03</td> <td>Supervisar la descarga y procesamiento de datos meteorológicos mediante logs.</td> </tr> </table>	T02	Validar los datos meteorológicos con los controles de calidad establecidos por la OMM.	T03	Supervisar la descarga y procesamiento de datos meteorológicos mediante logs.		
T02	Validar los datos meteorológicos con los controles de calidad establecidos por la OMM.						
T03	Supervisar la descarga y procesamiento de datos meteorológicos mediante logs.						
HU02	<table border="1"> <tr> <td>T01</td> <td>Implementar el algoritmo de detección de heladas y emisión de alertas.</td> </tr> <tr> <td>T02</td> <td>Programar una tarea en cron para la ejecución de la aplicación cada 1 minuto.</td> </tr> </table>	T01	Implementar el algoritmo de detección de heladas y emisión de alertas.	T02	Programar una tarea en cron para la ejecución de la aplicación cada 1 minuto.		
T01	Implementar el algoritmo de detección de heladas y emisión de alertas.						
T02	Programar una tarea en cron para la ejecución de la aplicación cada 1 minuto.						
HU03	<table border="1"> <tr> <td>T01</td> <td>Agregar los contactos que recibirán las alertas al servicio de mensajería del HGPT.</td> </tr> <tr> <td>T02</td> <td>Desarrollar un cliente REST para consumir el servicio de mensajería del HGPT.</td> </tr> <tr> <td>T03</td> <td>Registrar los mensajes SMS enviados a los suscriptores.</td> </tr> </table>	T01	Agregar los contactos que recibirán las alertas al servicio de mensajería del HGPT.	T02	Desarrollar un cliente REST para consumir el servicio de mensajería del HGPT.	T03	Registrar los mensajes SMS enviados a los suscriptores.
T01	Agregar los contactos que recibirán las alertas al servicio de mensajería del HGPT.						
T02	Desarrollar un cliente REST para consumir el servicio de mensajería del HGPT.						
T03	Registrar los mensajes SMS enviados a los suscriptores.						
HU04	<table border="1"> <tr> <td>T01</td> <td>Incorporar una sección para el SATH en la ficha de una estación meteorológica.</td> </tr> <tr> <td>T02</td> <td>Crear una interfaz gráfica web para mostrar los registros de heladas en una tabla.</td> </tr> <tr> <td>T03</td> <td>Generar una gráfica de múltiple eje del comportamiento climatológico de un evento de helada.</td> </tr> </table>	T01	Incorporar una sección para el SATH en la ficha de una estación meteorológica.	T02	Crear una interfaz gráfica web para mostrar los registros de heladas en una tabla.	T03	Generar una gráfica de múltiple eje del comportamiento climatológico de un evento de helada.
T01	Incorporar una sección para el SATH en la ficha de una estación meteorológica.						
T02	Crear una interfaz gráfica web para mostrar los registros de heladas en una tabla.						
T03	Generar una gráfica de múltiple eje del comportamiento climatológico de un evento de helada.						
HU05	<table border="1"> <tr> <td>T01</td> <td>Incorporar una interfaz gráfica web para ajustar los parámetros del detector de heladas .</td> </tr> <tr> <td>T02</td> <td>Integrar dos Input Spinner para cambiar los umbrales de temperatura en la interfaz web de ajuste de parámetros del detector de heladas.</td> </tr> <tr> <td>T03</td> <td>Integrar dos TimePicker para cambiar el horario de funcionamiento de monitoreo del clima en la</td> </tr> </table>	T01	Incorporar una interfaz gráfica web para ajustar los parámetros del detector de heladas .	T02	Integrar dos Input Spinner para cambiar los umbrales de temperatura en la interfaz web de ajuste de parámetros del detector de heladas.	T03	Integrar dos TimePicker para cambiar el horario de funcionamiento de monitoreo del clima en la
T01	Incorporar una interfaz gráfica web para ajustar los parámetros del detector de heladas .						
T02	Integrar dos Input Spinner para cambiar los umbrales de temperatura en la interfaz web de ajuste de parámetros del detector de heladas.						
T03	Integrar dos TimePicker para cambiar el horario de funcionamiento de monitoreo del clima en la						

		interfaz web de ajuste de parámetros del detector de heladas.
HU06	T01	Enlazar el servicio de mensajería del HGPT al proyecto del Geoportal.
	T02	Integrar un Autocomplete Tag Input para añadir o eliminar grupos de suscriptores en la interfaz web de ajuste de parámetros del detector de heladas.
HU07	T01	Parametrizar la obtención de datos meteorológicos para intervalos de registro de 1 y 5 minutos.
	T02	Rediseñar el formulario de creación o edición de una estación meteorológica.

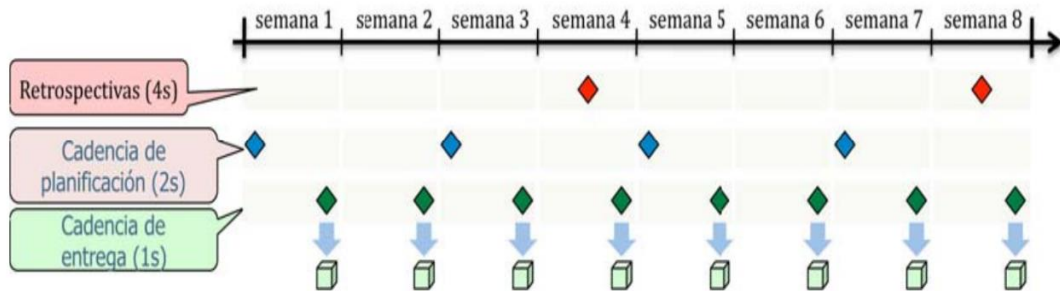
Elaborado por: El Investigador

Iteraciones

En Kanban no se prescriben iteraciones de tiempo fijo por lo que queda a criterio del equipo de desarrollo establecer el momento que se realizará la entrega, planificación y reunión retrospectiva para modificar y mejorar el proceso. Por lo tanto, se puede elegir realizar estas actividades de manera regular (“entrega cada dos semanas”) o bajo demanda (“la entrega cada vez que se tenga algo útil para entregar”).

Visto de esa forma, se podría utilizar iteraciones basadas en Scrum en la cual la planificación, retrospectiva y entrega se realizan en un tiempo fijo, determinando así una cadencia tal como se observa en la figura 5.

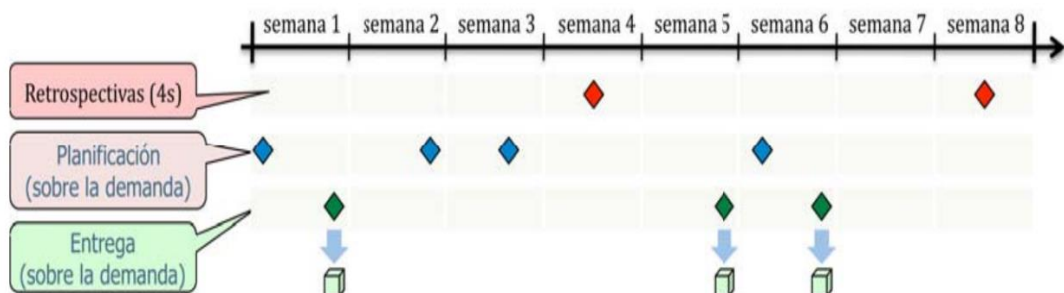
Fig. 5 Iteraciones basadas en Scrum



Fuente: C4Media Inc.

Por otra parte, se podría optar por iteraciones dirigidas por eventos tal como se observa en la figura 6. En este tipo de iteración, las reuniones de planificación se realizan al momento que se comienza a quedar sin cosas por hacer. Se lanza una entrega siempre y cuando exista un MMF listo por entregar. Las retrospectivas se realizan en un período de tiempo acordado y se puede efectuar un círculo de calidad espontáneo en el caso de que se tope con un mismo problema por segunda vez.

Fig. 6 Iteraciones dirigidas por eventos



Fuente: C4Media Inc.

Para el proyecto se decidió utilizar iteraciones dirigidas en eventos debido a que el proyecto se centra en la entrega de tres productos definidos en un acuerdo con el cliente, los productos son los siguientes:

- Publicación de datos meteorológicos en el Geoportal.

- Envío de mensajes SMS en caso de un evento de helada.
- Administración de los parámetros del detector de helada desde el Geoportal.

Atendiendo a estas consideraciones se estableció los siguientes criterios para la planificación, retrospectiva y entrega:

- La entrega se realizará al momento de que se tenga listo un MMF.
- La planificación se realizará al día siguiente de la entrega de un MMF.
- La retrospectiva se efectuará cada 2 semanas.

Tablero Kanban

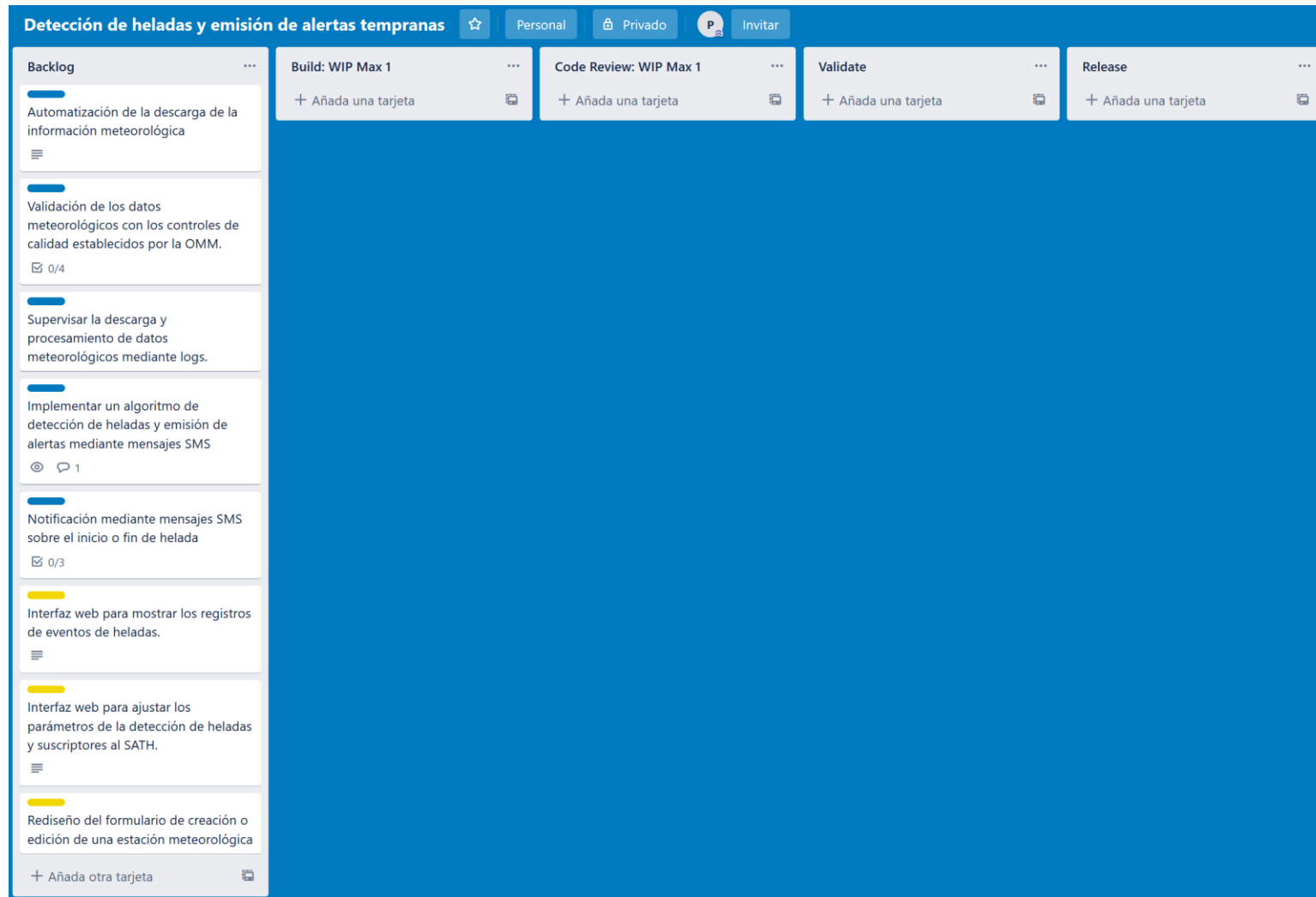
Para terminar, se procedió a crear el tablero Kanban en Trello con el flujo de trabajo de la figura 4. Las tareas se establecieron inicialmente en base a las historias de usuario de la tabla 6.

Además, se estableció en 1 el límite de trabajo en proceso o Work In Progress (WIP) para los estados de Build y Code Review debido a que solo habrá un solo persona programando.

Por otro lado, para diferenciar el tipo de tarea que se está realizando se estableció 3 etiquetas de colores. El color azul significa tareas vinculados con el procesamiento de la información, el color amarillo para la creación de las interfaces web y el color rojo para los bugs.

Finalmente, la estructura del tablero se estableció en un inicio tal como se muestra en la figura 7.

Fig. 7 Tablero Kanban



Elaborado por: El Investigador

3.1.3 Descripción de la base de datos

El sistema de información que maneja el HGPT para procesar los datos hidrometeorológicos enviados por las estaciones con transmisión en tiempo real o subidos manualmente por el administrador de red centraliza la información en una serie de colecciones en MongoDB.

Las colecciones utilizan el prefijo *red* para diferenciarlas de las demás colecciones y la estructura se encuentra definida en el proyecto de investigación “Sistema web de control, validación y publicación de datos de la red hidrometeorológica de Tungurahua para el Honorable Gobierno Provincial de Tungurahua”. [30]

Para el trabajo de investigación se utilizó las siguientes colecciones:

- *red.estaciones*, para obtener la información de las estaciones meteorológicas y los umbrales de los parámetros meteorológicos.
- *red.parametro*, para obtener los umbrales de los controles de consistencia temporal y de consistencia interna de los parámetros meteorológicos.
- *red.datos_meteorologia_1min*, para almacenar la información climatológica de datos instantáneos de 1 minuto.
- *red.carga_datos*, para registrar la subida de datos.

Considerando que se va a agregar la característica de detección de heladas y emisión de alertas a las estaciones meteorológicas se añadió los siguientes campos a la colección *red.estaciones*.

- *detectar_heladas* (Boolean) Indica si la estación meteorológica puede detectar fenómenos meteorológicos de heladas.
- *emitir_alertas* (Boolean) Indica si se debe enviar un mensaje SMS cuando inicie o finalice un fenómeno meteorológico de helada.
- *intervalo_registro* (String) Intervalo de tiempo en la que la estación meteorológica registrado los datos. Puede ser *1min* o *5min*.
- *conf_sath* (Object) Documento incrustado con la información de los parámetros de configuración para la detección de heladas y suscriptores al SATH.

Con respecto al documento incrustado *conf_sath* su estructura es la siguiente:

- *temperatura_inicio_alerta* (Double) Valor de temperatura para enviar la alerta de inicio de un posible evento de helada.
- *temperatura_fin_alerta* (Double) Valor temperatura para enviar la alerta de fin de helada.
- *hora_inicio_monitoreo* (String) Hora inicio del detector de heladas.
- *hora_fin_monitoreo* (String) Hora fin del detector de heladas.
- *suscribers* (Array of int) Lista de identificadores (claves primarias) de grupos suscriptos al SATH.

En cuanto a el campo *suscribers*, la información va a hacer referencia a las claves primarias de una tabla *GRUPOS*, perteneciente a un esquema con nombre *OFICIOS* de una base de datos Oracle.

El HGPT utiliza la base de datos Oracle para centralizar la información de su servicio de mensajería. De las tablas utilizadas por el servicio, *GRUPOS* y *CONTACTOS* tienen una relación directa con el proyecto debido a que alojaran la información de los suscriptores al SATH.

Hay que mencionar, además que las tablas *GRUPOS* y *CONTACTOS* que se muestra en la figura 8 mantienen una relación de muchos a muchos, por lo que utilizan una tabla adicional con nombre *GRUPOS_CONTACTOS* para romper la relación.

Fig. 8 Modelo relacional de las tablas Grupos y Contactos



Elaborado por: El Investigador

Para almacenar la información climatológica de datos instantáneos de 5 minutos se creó la colección *red.datos_meteorologia_5min* con los mismos campos de la colección *red.datos_meteorologia_1min*, pero con la diferencia de que se agregó campos para los valores máximo y mínimos de un parámetro meteorológico.

Campos fijos

- *_id* (ObjectId) Clave primaria del dato.
- *fecha_dato* (Date) Fecha de registro del dato por la estación.
- *fecha_subida* (Date) Fecha de carga del dato al sistema.
- *fecha* (String) Alias para la fecha en texto (optimizar búsquedas).
- *id_estacion* (ObjectId) Clave foránea de la colección *red.estaciones*.

Campos dinámicos

Por cada parámetro meteorológico asignado a la estación se crearán 6 campos con el prefijo del parámetro, obtenido de la colección *red.parametros*. Además, se incluirá los valores máximos y mínimos de cada parámetro meteorológico.

- *parametro* (Double) Valor del dato (promedio, mínimo o máximo).
- *parametro_estado* (String) Estado del valor después de ser validado (dudoso | erróneo)
- *parametro_cu* (Double) Valor del dato que no pasó el control de umbrales.
- *parametro_ctmin* (Double) Valor del dato que no pasó el control de consistencia temporal mínima.
- *parametro_ctmax* (Double) Valor del dato que no pasó el control de consistencia temporal máxima.
- *parametro_ci* (Double) Valor del dato que no pasó el control de consistencia interna.

La información generada durante un evento de helada se almacenará en la colección *red.eventos*, la cual tiene la siguiente estructura:

- *_id* (ObjectId) clave primaria.

- *inicio_alerta* (Boolean) estado de la alerta. True si se emitió alerta de inicio de helada, caso contrario False.
- *fecha_inicio_alerta* (Date) fecha en la que se emitió la alerta de inicio de helada.
- *alerta_fin* (Boolean) estado de la alerta. True si se emitió alerta de fin de helada, caso contrario False.
- *fecha_fin_alerta* (Date) fecha en la que se emitió la alerta de fin de helada.
- *duracion* (32-bit integer) segundos transcurridos durante evento de helada.
- *id_estacion* (ObjectId) Clave foránea de la colección ***red.estaciones***.

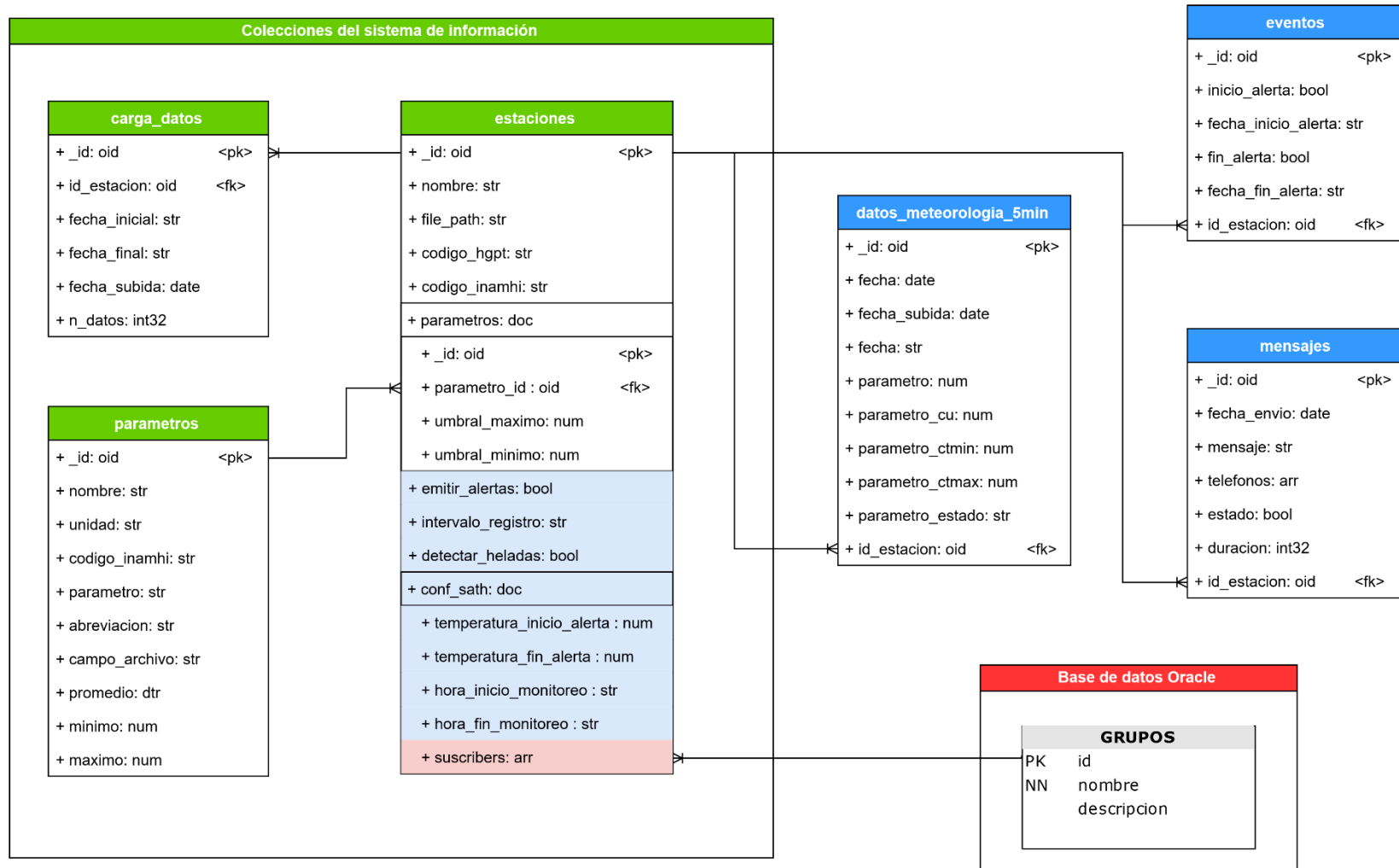
Con respecto a las alertas emitidas mediante mensajes SMS solo se registrará los envíos realizados en la colección ***red.sat.mensajes***. La colección presenta la siguiente estructura:

red.sat.mensajes

- *_id* (ObjectId) Clave primaria.
- *fecha_envio* (Date) Fecha de envío del mensaje.
- *mensaje* (String) El contenido del mensaje.
- *telefonos* (Array) Lista de teléfonos a cuáles se envió el mensaje.
- *estado* (Boolean) Estado del mensaje. True si se ha enviado, caso contrario False.
- *id_estacion* (ObjectId) Clave foránea de la colección ***red.estaciones***.

Por último, se creó el esquema que se muestra en la figura 9 para describir la estructura de las colecciones.

Fig. 9 Estructura de la base de datos



Elaborado por: El investigador

3.1.4 Configuración del proyecto

Archivo de configuración

Dado que la aplicación va a requerir conectarse a varios servicios es necesario mantener la información de autenticación en algún sitio. Una buena práctica es almacenar la información sensible en archivos de configuración debido a que brinda mayor seguridad y permite realizar cambios fácilmente. Otra ventaja de utilizar los archivos de configuración es que se puede especificar el comportamiento de la aplicación en entornos de desarrollo o producción.

Para el proyecto se eligió utilizar un archivo en formato JSON con la estructura que se muestra en la figura 10. El archivo será cargado automáticamente al inicio de la ejecución de la aplicación.

Fig. 10 Estructura de los archivos de configuración

```
{
  "ftp": {
    "host": "",
    "port": "",
    "username": "",
    "password": ""
  },
  "mongodb": {
    "host": "",
    "port": "",
    "username": "",
    "password": "",
    "db": ""
  },
  "smsapi": {
    "url": "",
    "username": "",
    "password": ""
  },
  "default_timezone": "America/Guayaquil",
  "maxfiles1min": 1,
  "maxfiles5min": 1
}
```

Elaborado por: El Investigador

Sistema de logs

En este tipo de software es muy importante tener un registro de como fue el comportamiento durante la ejecución, por lo que se implementó un sistema de logs con la librería nativa de Python llamada *logging*.

La forma como se visualizarán los logs dependerá del entorno en el que se encuentre ejecutando, si está en desarrollo se mostrarán los logs en consola y en archivos, mientras que si está en producción solo se guardará en archivos los cuales se almacenaran en la carpeta logs localizada en la raíz del proyecto.

Estos parámetros de configuración se deberán definir cuando inicie la aplicación, de forma que esté disponible lo antes posible para los demás módulos. Para cumplir esta restricción se definió la función que se muestra en el siguiente fragmento de código.

```
1. def setup_logger():
2.     """
3.     Establece la configuración para los logs.
4.     """
5.
6.     global ENV
7.
8.     # Formato del log
9.     logfmt = '%(asctime)s — %(name)s — %(levelname)s — %(message)s'
10.    datefmt = '%Y-%m-%d %H:%M:%S'
11.    default_formatter = logging.Formatter(logfmt, datefmt=datefmt)
12.
13.    # Ruta de la carpeta donde se guardarán los archivos de logs.
14.    log_file = f'{LOG_PATH}/logs'
15.
16.    console_handler = logging.StreamHandler(sys.stdout)
17.    console_handler.setFormatter(default_formatter)
18.    console_handler.setLevel(logging.DEBUG)
19.    file_handler = logging.handlers.TimedRotatingFileHandler(log_file, when='midnight')
20.    file_handler.setFormatter(default_formatter)
21.    file_handler.setLevel(logging.INFO)
22.    logging.root.setLevel(logging.DEBUG)
23.
24.    if ENV is 'PRODUCTION':
25.        logging.root.addHandler(file_handler)
26.    else:
27.        logging.root.addHandler(console_handler)
28.        logging.root.addHandler(file_handler)
```

Para utilizar los logs se definió una instancia en cada módulo del proyecto en Python como se muestra en el fragmento de código.

```
1. import logging
2.
3. logger = logging.getLogger(__name_)
4. logger.info('Working with logs !')
```

El resultado del anterior fragmento de código será el log que se muestra en la figura 11.

Fig. 11 Ejemplo de log

```
2018-11-16 11:56:40 — core — INFO — Working with Logs !
```

Elaborado por: El Investigador

3.1.5 Procesamiento de datos

Mapeo de entidades

MongoDB almacena la información en documentos, agrupados en colecciones que pertenecen a una base de datos. Para realizar las operaciones CRUD desde Python por lo regular se utiliza la librería oficial publicada por los desarrolladores de MongoDB llamada PyMongo. Esta librería es muy fácil de usar, sin embargo, se tendrá que escribir mucho código para realizar las operaciones CRUD, sobre todo en proyectos grandes.

Por esta razón se decidió utilizar MongoEngine debido a que proporciona mayor abstracción sobre PyMongo. La abstracción que proporciona MongoEngine está basada en clases, básicamente lo que realiza es mapear el esquema de las colecciones de MongoDB a clases, de esta forma se puede realizar las operaciones CRUD como si se manejara objetos.

Para el proyecto se mapeó las colecciones definidas en el diagrama de la figura 7 a las clases que se presentan en la tabla 8.

Tabla 8 Mapeo de colecciones a entidades MongoEngine

Clases	Colección
WeatherStation	red.estaciones
WeatherStationParameter	

WeatherParameter	red.parametro
MeteorologicalData5Min	red.datos_meteorologia_1min
MeteorologicalData1Min	red.datos_meteorologia_1min
Upload	red.carga_datos
Event	red.sat.eventos
SMS	red.sat.mensajes

Elaborado por: El Investigador

Así, por ejemplo, en el caso de la colección *red.estaciones* se creó la clase que se muestra en el siguiente fragmento de código.

```

1. import mongoengine
2.
3. class WeatherStation(mongoengine.Document):
4.     # Atributos
5.

```

En la línea 3, la clase descende una clase llamada *Document* perteneciente a *MongoEngine*. Esta declaración siempre se debe realizar en el caso de que no se quiera hacer dinámico la estructura del documento en *MongoDB*, caso contrario se debería descender de la clase *DynamicDocument*.

Respecto a los atributos, *MongoEngine* brinda una serie de clases que referencian a los tipos de datos que soporta *BSON* incluyendo una serie de propiedades que sirven para validar el atributo. Por lo que simplemente se debe especificar el atributo con el correspondiente tipo de dato *BSON* establecido en el campo de la colección, tal como muestra en el fragmento de código.

```

1. class WeatherStation(mongoengine.Document):
2.     """
3.     Clase enlazada a la colección **red.estaciones**
4.     """
5.     name = mongoengine.StringField(db_field='nombre', required=True)
6.     ftppath= mongoengine.StringField(db_field='file_path')
7.
8.     .....

```

Para más información ir a la documentación oficial de *MongoEngine* en el siguiente enlace: docs.mongoengine.org

Automatización de la descarga de datos del servidor FTP

La información climatológica enviada por las estaciones con transmisión de datos en tiempo real se almacena temporalmente en un servidor FTP del HGPT. Para descargar estos archivos se desarrolló un cliente FTP en Python, el cual realiza las siguientes tareas:

- Abrir una conexión vía FTP con el servidor **ftp.tungurahua.gob.ec**.
- Lista la carpeta de la estación para luego guardar los nombres de los archivos en una lista ordenados de antiguo a reciente. Si un archivo está siendo escrito o esta vacío se ignora.
- Lee cada archivo de la lista, lo convierte a un diccionario, quita la información innecesaria para finalmente enviarlo a que se procese.
- Una vez leído mueve el archivo del servidor FTP a un directorio llamado *saved*.
- En el caso de que el contenido del archivo sea información errónea se moverá a la carpeta *damaged*.

Validación de datos meteorológicos

Los datos meteorológicos registrados por las estaciones meteorológicas automáticas tienen que ser validados por una serie de controles de calidad definidos por la OMM.

El documento “Guidelines on Quality Control Procedures for Data from Automatic Weather Stations” [31] de la OMM establece aplicar los siguientes controles a los datos meteorológicos.

- Control de umbrales
- Control de consistencia temporal
 - Variabilidad máxima permitida de valores instantáneos
 - Variabilidad mínima requerida de valores instantáneos
- Control de consistencia interna

La aplicación de estos controles dependerá del tipo del parámetro meteorológico, así pues, en la tabla 9 se muestra los controles que debe realizar cada parámetro meteorológico.

Tabla 9 Controles de calidad aplicados a los parámetros meteorológicos

Parámetro Meteorológico	Umbrales	Consistencia temporal		Consistencia interna
		Variabilidad máxima permitida	Variabilidad mínima requerida	
Temperatura del Aire	X	X	X	
Humedad Relativa	X	X	X	
Velocidad del Viento	X	X	X	X
Dirección del Viento	X		X	X
Precipitación	X			
Punto de Rocío	X	X	X	X

Elaborado por: El Investigador

Control de umbrales

Verifica que los valores instantáneos estén dentro de un rango de límites aceptables. En el caso de que los valores estén fuera de los umbrales se marcará como erróneo. En el fragmento de código se muestra la función que realiza este control.

```

1. def check_thresholds(value, _min, _max):
2.     """
3.     Verifica que un dato meteorológico este dentro de un rango de límites aceptables
4.
5.     :param decimal.Decimal value: dato meteorológico
6.     :param decimal.Decimal _min: umbral mínimo
7.     :param decimal.Decimal _max: umbral máximo
8.     :return: True si paso el control, caso contrario False
9.     :rtype: bool
10.    """
11.    return _min <= value <= _max

```

Control de consistencia temporal

Detecta saltos de valores o repeticiones irrealistas debido a sensores bloqueados. El control se divide en dos nuevos controles: control de variabilidad máxima permitida y control de variabilidad mínima requerida.

El **control de variabilidad máxima permitida** verifica que un valor instantáneo no difiera demasiado del valor anterior. En el caso de que el valor instantáneo supere el

límite establecido con el valor anterior se marcará como dudoso. En el fragmento de código se muestra la función que realiza la validación.

```
1. def check_maximum_allowed_variability (weather, param, ctmax, seconds=60):
2.     """
3.     Verifica que un parámetro meteorológico cumpla con el control
4.     de variabilidad máxima permitida
5.
6.     :param models.Weather weather: información climatológica
7.     :param str param: nombre del campo (parámetro meteorológico)
8.     :param decimal.Decimal ctmax: umbral de variabilidad máxima permitida
9.     :param int seconds: segundos para el valor anterior
10.    :return: True si paso el control, caso contrario False
11.    :rtype: bool
12.    """
13.    enddt = weather.dt
14.    startdt = helpers.subtract_seconds(enddt, seconds)
15.    previous = db.getweather(enddt, startdt)
16.
17.    if previous:
18.        prev = previous.__getitem__(param)
19.        current = weather.__getitem__(param)
20.        diff = abs(current - prev)
21.
22.        if diff > ctmax:
23.            return False
24.
25.    return True
```

El **control de variabilidad mínima requerida** verifica que el valor instantáneo cambie durante un tiempo máximo de 1 hora. En el caso de que el valor se mantenga por una hora se marcará como dudoso. En el fragmento de código se muestra la función que realiza la validación.

```
1. def check_minimum_required_variability(weather, param, ctmin, mindata):
2.     """
3.     Verifica que un parámetro meteorológico cumpla con el control
4.     de variabilidad mínima requerida
5.
6.     :param models.Weather weather: información climatológica
7.     :param str param: nombre del campo (parámetro meteorológico)
8.     :param decimal.Decimal ctmin: umbral de variabilidad mínima requerida
9.     :param int mindata: número de datos mínimos para realizar el control
10.    :return: True si paso el control, caso contrario False
11.    :rtype: bool
12.    """
13.    isvalid = False
14.    enddt = weather.dt
15.    startdt = helpers.subtract_seconds(enddt, 3600)
16.    previous = db.getweather(weather.station_id, startdt, enddt)
17.
18.    if previous.count() > mindata:
19.        for prev in previous:
20.            prev = prev.__getitem__(param)
21.            current = weather.__getitem__(param)
22.            diff = abs(current - prev)
23.            if diff >= ctmin:
24.                isvalid = True
```

```

25.         break
26.     else:
27.         isvalid = True
28.
29.     return isvalid

```

Control de consistencia interna

El control de consistencia interna evalúa datos basándose en la relación entre dos parámetros climáticos. En el caso de las estaciones meteorológicas por el momento solo se aplica este control a la velocidad del viento, dirección del viento y punto de rocío. Estos parámetros deben cumplir lo siguiente:

- Si la velocidad del viento es cero, la dirección de viento también debe ser cero.
- Si la velocidad del viento es diferente a cero, la dirección de viento también debe ser diferente a cero.
- El punto de rocío debe ser menor o igual a la temperatura del aire.

Para la dirección y velocidad del viento se creó la función que se muestra en el siguiente fragmento de código.

```

1. def check_internal_consistency_ws(self, ws, wd):
2.     """
3.     Verifica que la velocidad del viento cumpla con el control de consistencia interna
4.
5.     :param decimal.Decimal ws: velocidad de viento
6.     :param decimal.Decimal wd: dirección de viento
7.     :return: True si paso el control, caso contrario False
8.     :rtype: bool
9.     """
10.    return (ws == 0 and wd != 0) or (ws != 0 and wd == 0)

```

En cambio, para el punto de rocío se creó la función que se muestra en el siguiente fragmento de código.

```

1. def check_internal_consistency_td(self, t, td):
2.     """
3.     Verifica que el punto de rocío cumpla con el control de consistencia interna
4.
5.     :param decimal.Decimal t: temperatura del aire
6.     :param decimal.Decimal td: punto de rocío
7.     :return: True si paso el control, caso contrario False
8.     :rtype: bool
9.     """
10.    return td <= t

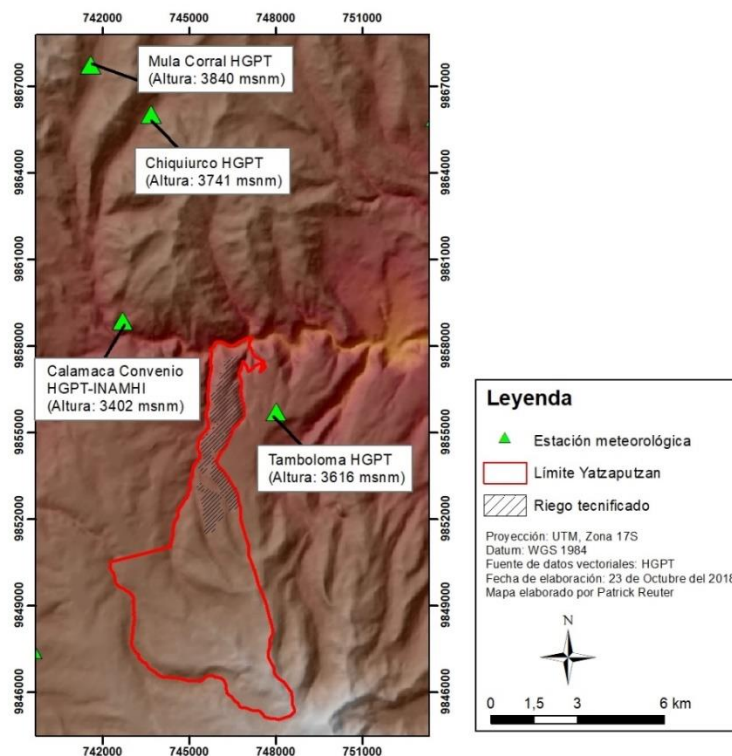
```

3.1.6 Detección anticipada de un evento de helada

El estudio de la detección anticipada de un evento de helada fue realizado por el M.Sc. Patrick Reuter, consultor contratado por la GIZ para definir el proceso de detección de heladas y emisión de alertas tempranas vinculado con la estación meteorológica Tamboloma.

El consultor trabajó con la estación meteorológica de Tamboloma como estación principal debido a que es considerada como la más representativa para medir el clima de la zona media de riego tecnificado de Yatzaputzan, sitio donde se realiza el proyecto piloto de Sistema de Alerta Temprana. Además, el consultor utilizó las estaciones meteorológicas Mula Corral, Chiquiurco y Calamaca como estaciones de apoyo. Estas estaciones se encuentran al norte de la zona de intervención como se observa en la figura 12. También cuentan con sensores para medir la dirección y velocidad del viento en contraste con la estación de Tamboloma que solo tiene instalada sensores para temperatura, humedad relativa y precipitación.

Fig. 12 Estaciones meteorológicas utilizadas en el estudio de detección anticipada de una helada



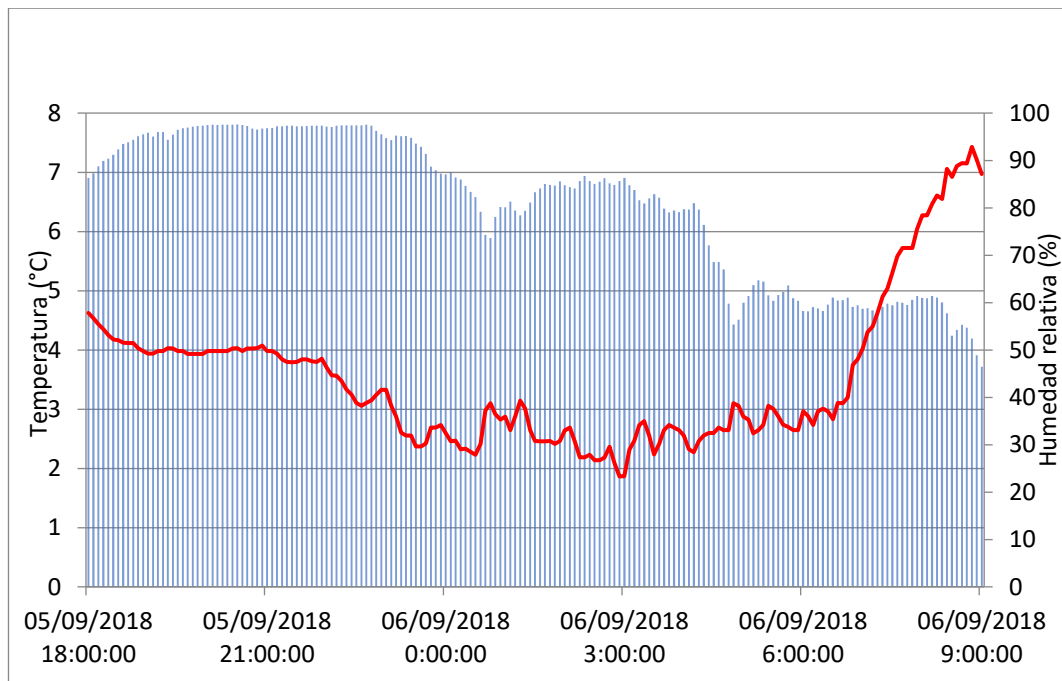
Elaborado por: Patrick Reuter

Patrones meteorológicos antes y durante las heladas

El estudio comenzó con un análisis de heladas reportadas por la población local en los meses de septiembre y octubre de 2018 con la finalidad de distinguir entre las condiciones climáticas que forman este evento extremo y aquellas de una noche normal.

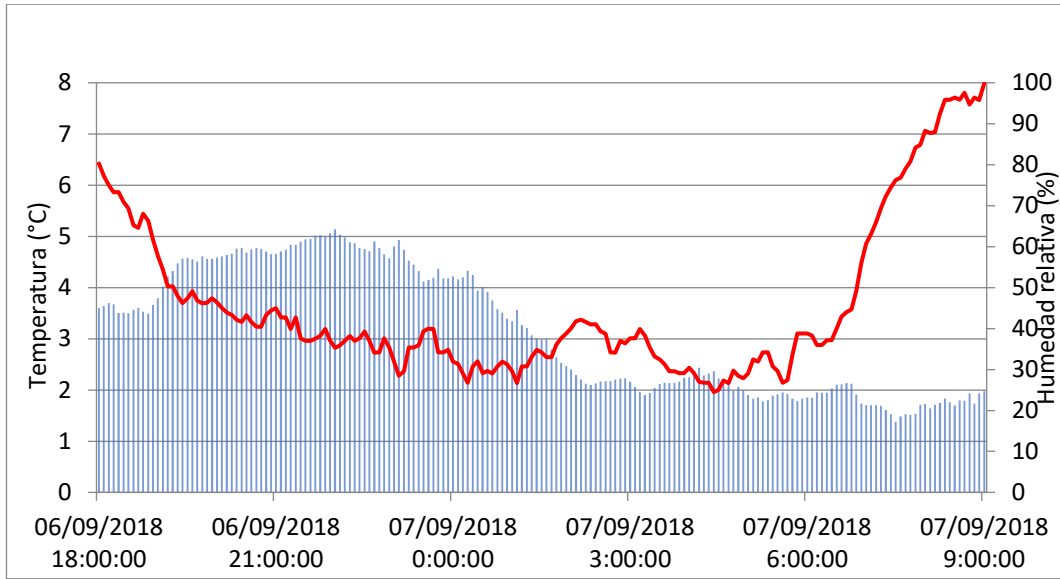
Según la población local, en las madrugadas del 6 y 7 de septiembre de 2018 se presentaron dos heladas consecutivas, de las cuales especialmente la segunda causó grandes pérdidas en los cultivos, sobre todo de papa. Como se puede observar en las figuras 13 y 14, la temperatura bajó en ambas noches por debajo de 3°C , oscilando entre 3 y 2°C hasta aproximadamente las 07:00 de la mañana. La razón por la que la segunda fue más perjudicial es la baja humedad relativa del aire.

Fig. 13 Evento de helada en Tamboloma en la madrugada del 06/09/2018.



Elaborado por: Patrick Reuter

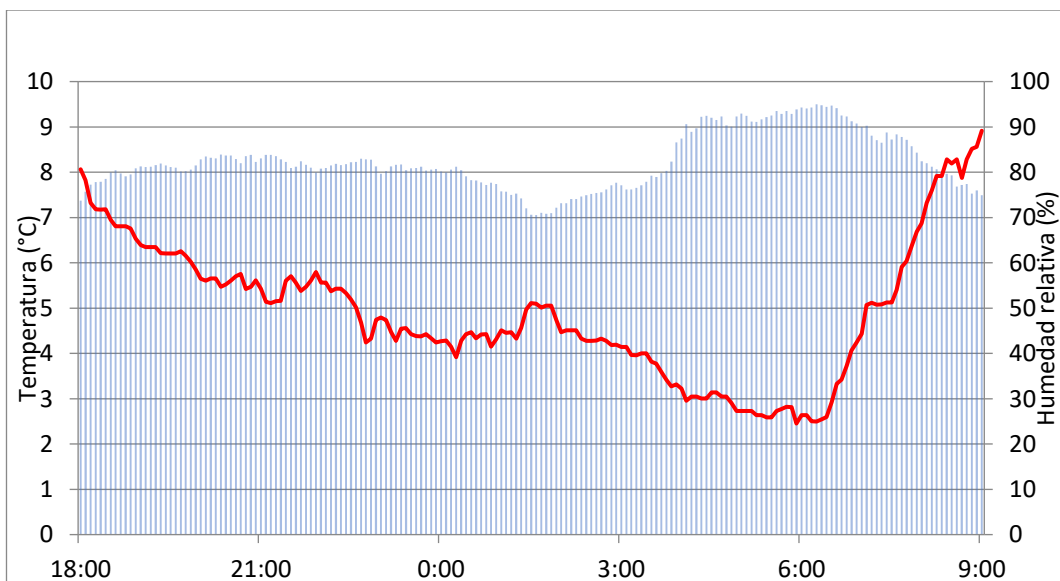
Fig. 14 Evento de helada en Tamboloma en la madrugada del 07/09/2018.



Elaborado por: Patrick Reuter

Otra helada, aunque menos fuerte, fue reportada por la gente de Tamboloma en la madrugada del 19 de septiembre de 2018. Como se puede observar en la figura 15, la temperatura cayó por debajo de 3°C por unas dos horas. La humedad relativa era alta, probablemente la razón por la que esta helada blanca no causó grandes pérdidas

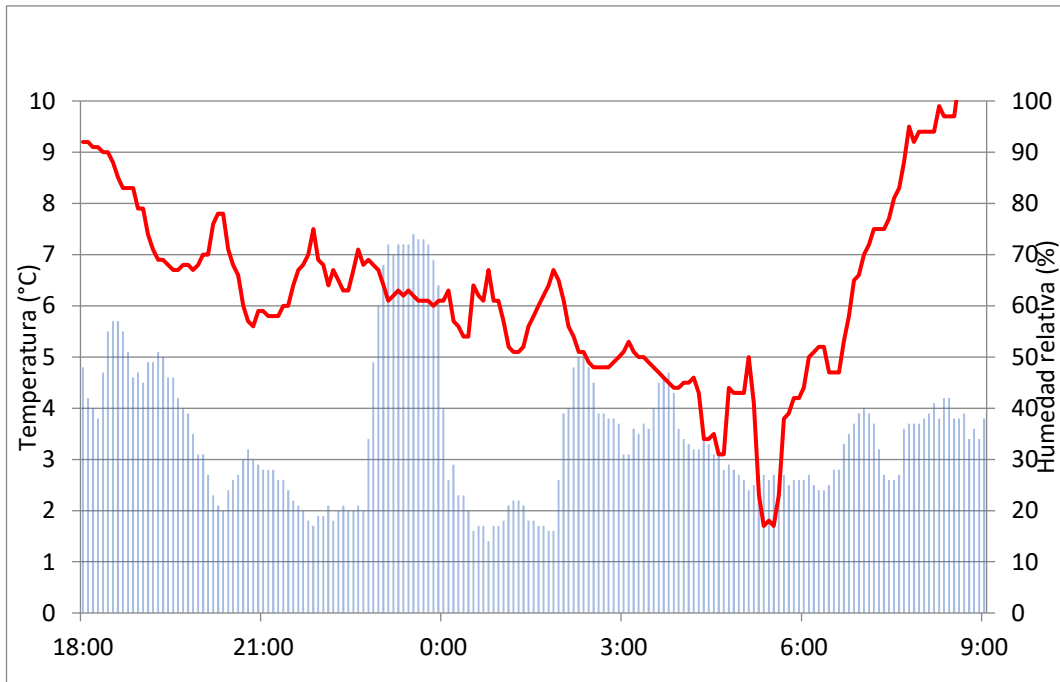
Fig. 15 Evento de helada en Tamboloma en la madrugada del 19/09/2018



Elaborado por: Patrick Reuter

En cambio, la helada del 24 de octubre de 2018 que se muestra en la figura 16 causó pérdidas significativas en los cultivos, ya que se trataba de una helada negra con baja humedad relativa. La temperatura bajó solamente un corto tiempo por debajo de 3°C, pero la caída fue fuerte.

Fig. 16 Evento de helada en Tamboloma en la madrugada del 24/10/2018



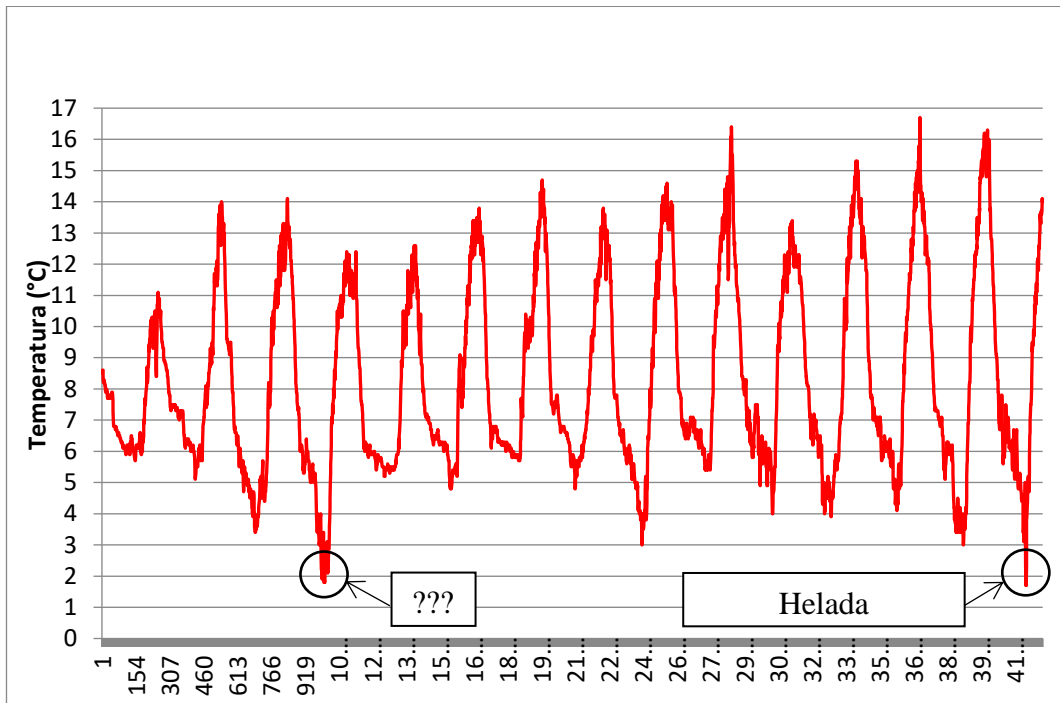
Elaborado por: Patrick Reuter

Patrones meteorológicos durante noches sin presencia de heladas

Después de haber analizado el comportamiento de temperatura y humedad relativa durante 4 heladas reportadas se procedió a analizar la temperatura durante las noches sin presencia de heladas.

En la figura 17 se muestra la temperatura mínima de cada 5 minutos en la estación Tamboloma desde el 9 de septiembre de 2018, 17:00 hasta el 24 de octubre de 2018, 11:40. Como se puede observar, la temperatura normalmente apenas cae por debajo de los 3°C en el periodo observado, esto pasó solamente dos veces, de las cuales la de la madrugada del 24 de octubre de 2018 fue confirmada como helada (véase en la figura 16).

Fig. 17 Temperatura mínima de cada 5 minutos en la estación Tamboloma desde el 09/10/2018 17:00 hasta el 24/10/2018 11:40



Elaborado por: Patrick Reuter

Conclusiones

Como resultado del estudio se realizó una serie de preguntas claves que se presentan en la tabla 10.

Tabla 10 Preguntas claves sobre los eventos de heladas en Tamboloma

<p>¿A partir de qué temperatura se presenta una helada?</p>	<p>La helada se presenta a partir de un umbral de 3°C</p>
<p>¿Cómo son las condiciones climáticas durante las heladas?</p>	<p>Mediante las estaciones de apoyo (Calamaca, Mula Corral, Chiquiurco) se puede concluir lo siguiente:</p> <ul style="list-style-type: none"> - No hay precipitación - Baja velocidad del viento (1-2 m/s)

	- Dirección del viento entre SE, S y SO
¿Cuándo termina la helada?	Durante una noche de helada, la temperatura puede ocasionalmente subir por las 3°C y después caer otra vez. Con un umbral máximo de 5°C durante la helada podemos capturar todo el periodo, quiere decir, cuando la temperatura sube por encima de 5°C, la helada terminó.
¿Por cuánto tiempo se presentan las heladas?	Con los umbrales anteriormente establecidos, el tiempo de las heladas es el siguiente: Helada del 06/09/18: 07h 35min Helada del 07/09/18: 10h 05min Helada del 19/09/18: 02h 25min Helada del 24/10/18: 20min

Elaborado por: Patrick Reuter

Algoritmo de detección de heladas y emisión de alertas tempranas

De las conclusiones realizadas sobre el estudio de la detección anticipada de un evento de helada se identificó que la temperatura desempeña un papel importante en la aparición de este fenómeno extremo. Ya que una caída de la temperatura por debajo de los 3°C supondría el inicio de la helada, la cual finalizaría hasta que la temperatura vuelva a subir.

Atendiendo a estas consideraciones, se estableció un umbral mínimo de temperatura para enviar la alerta de inicio de riesgo de helada y otro umbral para la finalización de

la alerta. Las alertas se emitirán mediante mensajes SMS a grupos de suscriptores asociados a la estación meteorológica.

Hay que mencionar, además que en el estudio se analizó la humedad relativa para determinar si una helada fue negra o blanca, por esta razón se incluyó al proceso de detección de helada junto con el punto rocío.

De los parámetros meteorológicos antes mencionados solo el punto de rocío no es registrado por las estaciones, por lo cual se decidió utilizar la siguiente fórmula para calcular la temperatura de rocío. [32]

$$H = \frac{\log_{10} RH - 2}{0.4343} + \frac{17.62 * T}{243.12 + T}$$
$$Dp = \frac{243.12 * H}{17.62 - H}$$

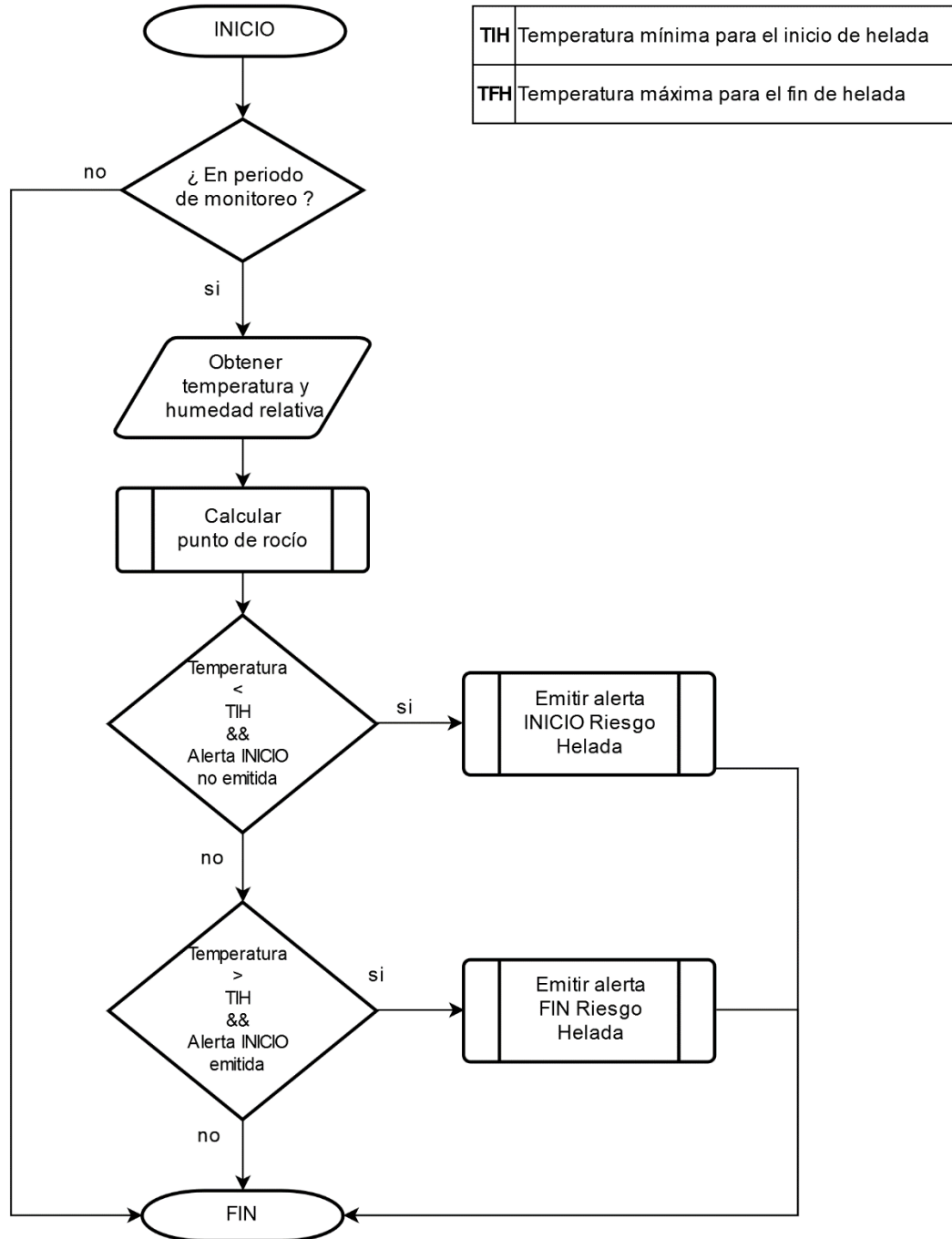
Donde T es la temperatura en grados Celsius y RH la humedad relativa.

En cuanto a el periodo de aparición de una helada se determinó que por lo regular se producen en las madrugadas, de modo que monitorear el clima en horas del día sería innecesario debido que la temperatura sube por encima del umbral de riesgo de helada. Para cumplir esta restricción se estableció una variable de tiempo para el monitoreo del clima.

En resumen, para la detección anticipada de un evento helada se va a monitorear la temperatura, humedad relativa y punto de rocío. Sin embargo, por el momento solo se va a evaluar y verificar que la temperatura cumpla con unos umbrales para la emisión de las alertas de inicio o fin de riesgo de helada. Este proceso se va a llevar a cabo en tiempo real y solo en periodos de tiempo definidos.

A partir de las premisas expuestas, se diseñó el algoritmo que se muestra en el diagrama de flujo de la figura 18 para la emisión de alertas de inicio o fin de un posible evento de helada.

Fig. 18 Algoritmo de detección de heladas y emisión de alertas tempranas



Elaborado por: El Investigador

Implementación del algoritmo

Para la implementación del algoritmo que se muestra en el diagrama de flujo de la figura 16 se creó una clase llamada *FrostDetector* con la finalidad de monitorear el clima en busca de posible evento de helada.

La estructura de la clase se muestra en el siguiente fragmento de código:

```
1. class FrostDetector:
2.     """
3.     Clase para monitorear el clima en busca de un evento de helada.
4.     """
5.     def __init__(self):
6.         """
7.         Iniciliza los atributos del objeto
8.         """
9.
10.    def start(self):
11.        """
12.        Monitorea los datos climatológicos
13.        """
14.
15.    def _emitalert(self):
16.        """
17.        Envía la alerta de INICIO o FIN de riesgo de helada
18.        """
19.
20.    def _buildmessage(self):
21.        """
22.        Genera los mensajes de INICIO o FIN de riesgo de helada
23.        """
```

Se debe agregar que la información climatológica enviada por las estaciones automáticas con transmisión de datos en tiempo debe ser validada antes de que entren al monitoreo. Solo se evaluarán los datos que no hayan sido marcados como erróneos, es decir deben pasar el control de umbrales.

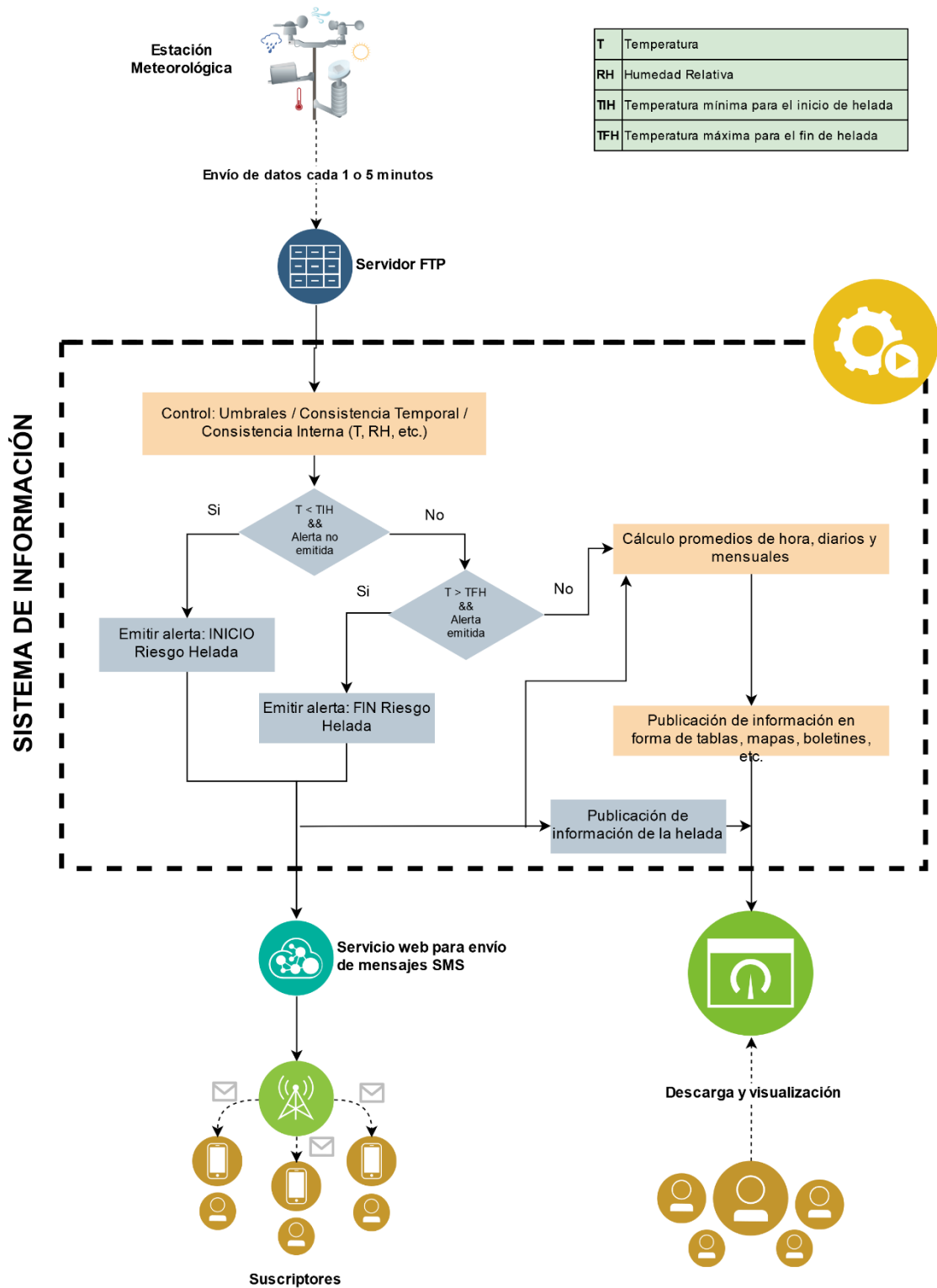
Con los datos validados se continuará con el proceso expuesto en el algoritmo del diagrama de flujo de la figura 16.

Ya sea que se emita la alerta de inicio o fin de riesgo de helada, o no, el proceso termina con las siguientes tareas.

- Reducción de la información climatológica en promedios hora y diarios.
- Publicación de la información climatológica en el Geoportal.
- Notificación al administrador del Geoportal en el sitio web acerca de las tareas efectuadas.

Teniendo en cuenta lo expuesto anteriormente, en la figura 19 se muestra el flujo completo de procesamiento de la información climatológica desde la obtención de los datos hasta la publicación en el sitio web.

Fig. 19 Proceso de detección de heladas y emisión de alertas tempranas



Elaborado por: El Investigador

3.1.7 Emisión de alertas mediante mensajes SMS

Al momento que los datos meteorológicos cumplan con las condiciones para el inicio o fin de un evento de helada se enviará un mensaje de texto a los suscriptores pertenecientes a la estación.

El contenido de los mensajes que van a recibir los suscriptores es el que se describe en la tabla 11.

Tabla 11 Contenido de los mensajes SMS

ALERTA DE INICIO	ALERTA DE FIN
HGPT- { estación } Informa: INICIO Riesgo de Helada. { temperatura }. { dd/mm/yyyy } { hh:mm }	HGPT- { estación } Informa: FIN Riesgo de Helada. { temperatura }. { dd/mm/yyyy } { hh:mm }

Elaborado por: El Investigador

Los campos que se están entre llaves {} son valores variables. Por lo tanto, el campo *estación* mostrará el nombre de la estación meteorológica. En cambio, el campo *temperatura* indicará el valor de la temperatura en grados Celsius al momento de una bajada o subida de temperatura. Por último, el campo *dd/mm/yyyy hh:mm* señalará la fecha y hora del envío de la alerta.

El contenido de los mensajes debe ser aprobado por las Operadoras Celulares, tarea que realiza la Dirección de Sistemas comunicando al contratista. Por otra parte, en el caso de la temperatura no se enviará el símbolo de grado (°) debido a las limitaciones de caracteres de los mensajes SMS.

Cliente REST

Actualmente el HGPT tiene en funcionamiento un servicio de mensajería para la comunicación de la institución mediante mensajes SMS. En la figura 20 se muestra la página de inicio de este servicio.

Fig. 20 Página de inicio del servicio de mensajería del HGPT



Fuente: HGPT

Para utilizar el servicio de mensajería, un usuario autorizado debe ingresar al sitio web que se presenta en la figura 20, una vez dentro enviará los mensajes ya sea a un contacto o grupos de contactos desde un formulario.

Otra alternativa, es mediante un servicio web de arquitectura REST, construido con la finalidad de ofrecer el servicio a terceros. El servicio web cuenta con un mecanismo de autenticación básica y expone las URIs que se muestra en la tabla 12.

Tabla 12 URIs del servicio web de envío de mensajes SMS

URIs	Tipo	Parámetros	Descripción
/send_sms	POST	- phones - message	Envía un mensaje de texto a una lista de números de teléfonos.
/send_group	POST	- group - message	Envía un mensaje de texto a un grupo de contactos.

Elaborado por: El Investigador

En ambas URIs, el parámetro *message* establece el contenido de los mensajes de texto. Por otro lado, el parámetro *phones* de la URI /send_sms indica los números de

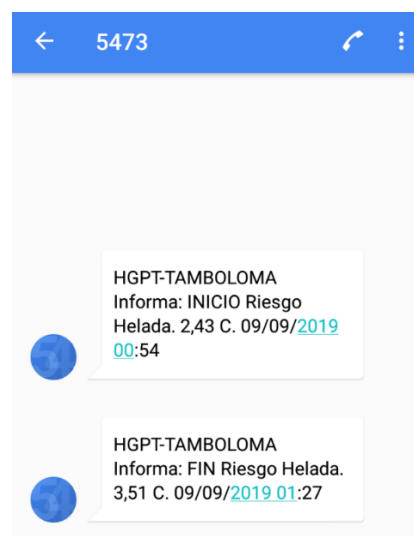
teléfonos de los destinatarios. Por último, el parámetro *group* de la URI */send_group* establece el identificador del grupo de destinatarios.

Para el proyecto se eligió utilizar el servicio web, específicamente para el envío de mensajes a grupos de destinatarios. Es así que se desarrolló un cliente REST en Python con la librería *requests*. En el fragmento de código se muestra la función de envío de mensajes.

```
1. def sendsms(msg, ids, user):
2.     """
3.     Envía un mensaje de texto a los grupos del SATH
4.
5.     :param str: mensaje
6.     :param list of int ids: claves primarias de la tabla oficinas.GRUPOS (Oracle)
7.     :param str user: usuario que envia los mensajes SMS.
8.     la estaciones meteorológicas
9.     """
10.    API_URL = settings.SMS['url']
11.    USERNAME = settings.SMS['username']
12.    PASSWORD = settings.SMS['password']
13.
14.    for groupid in ids:
15.        payload = {'group': groupid, 'message': msg, 'user': user}
16.        r = requests.post(F{API_URL}/send_group', data=payload, auth=(USERNAME, PASSWORD))
```

Los mensajes que recibirán los suscriptores serán similares a los que se muestra en la figura 21.

Fig. 21 Mensajes SMS de inicio y fin de riesgo de helada



Elaborado por: El Investigador

3.1.8 Publicación de datos

Al mismo tiempo que se emite la alerta de aparición de un evento de helada se debe publicar en la ficha de la estación la información sobre esta. Para cumplir este requisito se desarrolló un pequeño servicio web de arquitectura REST en CodeIgniter el cual expone las URIs que se muestra en la tabla 13.

Tabla 13 URIs del servicio web de heladas

URIs	Tipo	Parámetros	Descripción
/frost	GET	- station_id - from - to	Obtiene una lista de los eventos de heladas de una estación
/frost/weather	GET	- station_id - interval - from - to	Obtiene los valores de la temperatura, humedad relativa y punto de rocío de un evento de helada.

Elaborado por: El Investigador

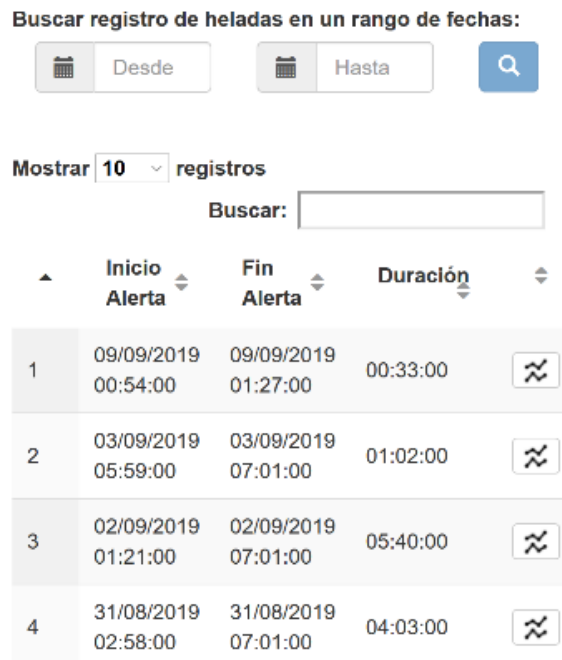
En ambas URIs la información de los parámetros es la siguiente:

- **station_id**, identificador de la estación meteorológica.
- **from**, fecha de inicio para una búsqueda.
- **to**, fecha de fin para una búsqueda.
- **interval**, intervalo de registro de datos de la estación meteorológica.

El servicio web se utilizó para obtener los registros de eventos de heladas e información climatológica desde una interfaz web integrada a la ficha de la estación.

La interfaz web es la que se muestra en la figura 22, cuenta con un listado de todas las alertas enviadas hasta la fecha. Además, permite realizar búsquedas y generar una gráfica como se observa en la figura 23 para visualizar el comportamiento de la temperatura, humedad relativa y punto de rocío durante el evento de helada.

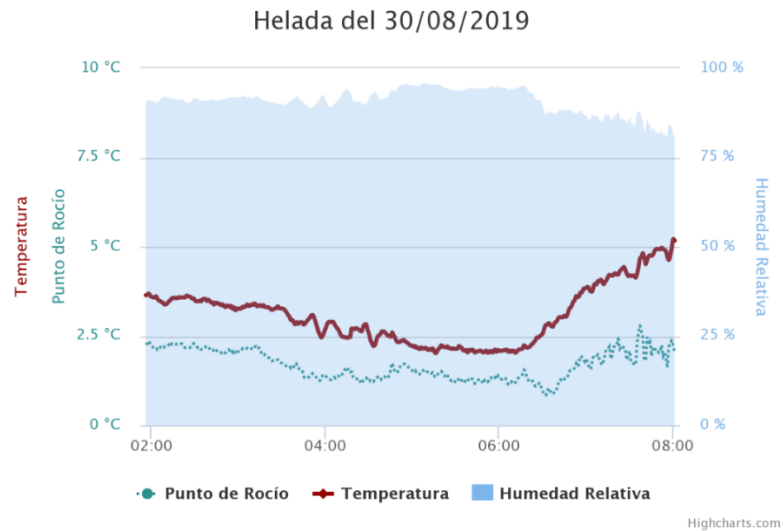
Fig. 22 Interfaz web de registros históricos de eventos de heladas.



Elaborado por: El Investigador

La información climatológica presentada al momento de generar la gráfica será de una hora antes de la emisión de la alerta de inicio de riesgo de helada y hasta una hora después de la alerta de fin de riesgo de helada.

Fig. 23 Gráfica de múltiple con la información de temperatura, humedad relativa y punto de rocío



Elaborado por: El Investigador

3.1.9 Configuración de parámetros de la emisión alertas

Para configurar los parámetros de umbrales de temperatura de inicio o fin de helada, periodo de monitoreo y suscriptores se diseñó la interfaz web que se muestra en la figura 24.

La interfaz web de configuración consta de tres partes:

- La parte superior izquierda contiene dos componentes, el primero permite cambiar el umbral mínimo de temperatura a la que se enviará la alerta de inicio de riesgo de helada, mientras que el segundo permite cambiar el umbral máximo de temperatura a la que se enviará la alerta de fin de riesgo de helada
- La parte superior derecha contiene dos componentes para cambiar la hora de inicio y fin del monitoreo del clima.
- La parte inferior contiene un campo con autocompletado para añadir o eliminar grupos que recibirán los mensajes SMS.

Fig. 24 Formulario de ajuste de parámetros

The image shows a web interface for configuring alert parameters. It consists of three main sections: 'Umbral de Temperatura', 'Monitoreo', and 'Suscriptores'. The 'Umbral de Temperatura' section has two sliders for 'Inicio' (set to 2.50 °C) and 'Fin' (set to 3.50 °C). The 'Monitoreo' section has two time pickers for 'Desde' (18:00) and 'Fin' (07:00). The 'Suscriptores' section has a search bar with the text 'alerta temprana tamboloma'. Below the sections is a blue 'Editar' button.

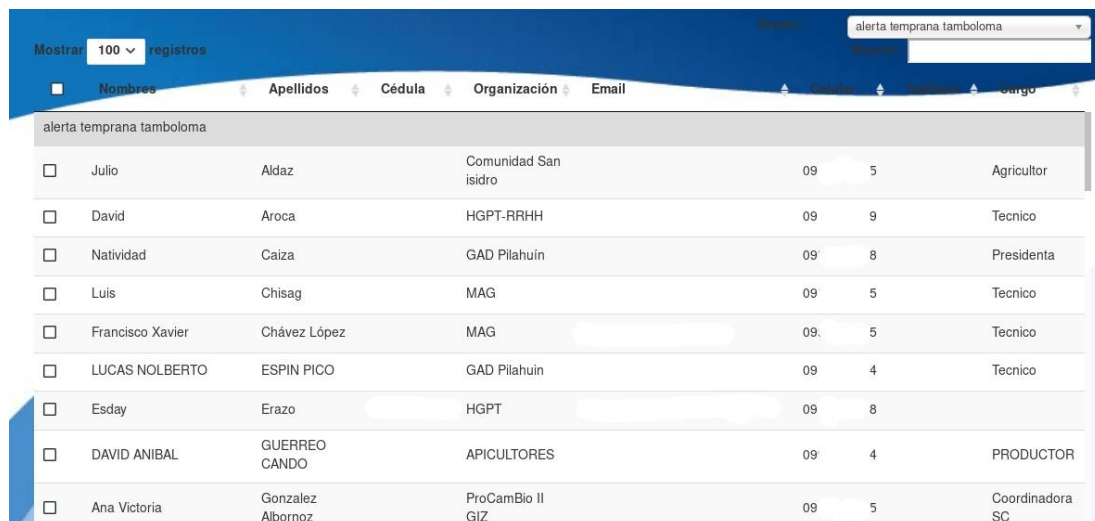
Elaborado por: El Investigador

Los grupos de suscriptores son obtenidos del servicio de mensajería del HGPT. De modo que para añadir más suscriptores el administrador deberá ingresar al sitio web del servicio de mensajería y crear los contactos.

La creación de contactos puede ser realizada de dos maneras, la primera es mediante un formulario y la segunda es con una plantilla en Excel que se cargará en el sitio web.

Para visualizar la información de los contactos el sitio web proporciona una sección para mostrar el listado de contactos pertenecientes a un grupo como se observa en la figura 25.

Fig. 25 Listado de suscriptores de la estación de Tamboloma



	Nombres	Apellidos	Cédula	Organización	Email	Teléfono	Número	Cargo
	alerta temprana tamboloma							
<input type="checkbox"/>	Julio	Aldaz		Comunidad San isidro		09	5	Agricultor
<input type="checkbox"/>	David	Aroca		HGPT-RRHH		09	9	Tecnico
<input type="checkbox"/>	Natividad	Caiza		GAD Pllahuin		09	8	Presidenta
<input type="checkbox"/>	Luis	Chisag		MAG		09	5	Tecnico
<input type="checkbox"/>	Francisco Xavier	Chávez López		MAG		09	5	Tecnico
<input type="checkbox"/>	LUCAS NOLBERTO	ESPIN PICO		GAD Pllahuin		09	4	Tecnico
<input type="checkbox"/>	Esday	Erazo		HGPT		09	8	
<input type="checkbox"/>	DAVID ANIBAL	GUERREO CANDO		APICULTORES		09	4	PRODUCTOR
<input type="checkbox"/>	Ana Victoria	González Albornoz		ProCamBio II GIZ		09	5	Coordinadora SC

Fuente: HGPT

3.1.10 Ejecución de pruebas

Para garantizar la calidad y correcto funcionamiento del software se realizó las siguientes pruebas:

- Pruebas Unitarias
- Pruebas Funcionales
- Pruebas de Aceptación

En el caso de las pruebas unitarias y funcionales se utilizó la librería **pytest**.

Pruebas Unitarias

Con las pruebas unitarias se comprobará el correcto funcionamiento de las partes más pequeñas de código (métodos y funciones). Para el proyecto se definieron los casos de prueba que se muestran en la tabla 14 y la ejecución se presenta en la figura 26.

Tabla 14 Casos de prueba para las funciones y métodos

Función / Método	Casos de prueba
Cálculo de punto de rocío	<ul style="list-style-type: none"> - Validar con los ejemplos mostrados en el documento. [29] - Comprobar que el tipo de dato devuelto sea <i>Decimal</i>. - Verificar que el valor devuelto tenga dos cifras significativas.
Conversión a fechas	<ul style="list-style-type: none"> - Comprobar que la cadena a convertir tenga el formato %y-%m-%d %H:%M:%S. - Verificar que la fecha devuelta se UTC. - Verificar que la fecha devuelta no tenga información de una zona horaria. - Verificar que la fecha devuelta tenga una zona horaria.
Conversión a decimal	<ul style="list-style-type: none"> - Comprobar que el dato a convertir sea un número real válido. - Validar la precisión. - Verificar las cifras significativas.
Obtención de información climatológica	<ul style="list-style-type: none"> - Comprobar datos faltantes - Verificar los campos vacíos - Validar que el tipo de dato que utiliza la información climatológica sea <i>Decimal</i>.
Almacenamiento y actualización de la información climatológica.	<ul style="list-style-type: none"> - Validar que se establezcan los estados de control para datos válidos, erróneos, dudosos e inconsistentes. - Verificar que las fechas de almacenamiento sea en UTC.
Obtención de parámetros meteorológicos	<ul style="list-style-type: none"> - Comprobar que un parámetro meteorológico tenga la información de umbrales de validación. - Verificar que el tipo de dato de los umbrales sea <i>Decimal</i>.

	- Verificar cuales controles debe realizar un parámetro meteorológico.
--	--

Elaborado por: El Investigador

Fig. 26 Ejecución de pruebas unitarias

```

===== test session starts =====
platform linux -- Python 3.6.8, pytest-4.6.3, py-1.8.0, pluggy-
0.12.0
rootdir: /home/developer/workspace/frostdetector/tests/unit
collected 18 items

test_models.py .. [ 30%]
test_dbqueries.py .. [ 70%]
test_helpers.py ... [100%]

===== 18 passed in 2.1 seconds =====

```

Elaborado por: El Investigador

Pruebas Funcionales

La finalidad de las pruebas funcionales es comprobar que el software funcione acorde a las especificaciones funcionales y requisitos del cliente.

Para el proyecto se validaron las siguientes funcionalidades y la ejecución se presenta en la figura 27.

- Descarga de la información climatológica desde el servidor FPT
- Control de umbrales
- Control de variabilidad máxima permitida
- Control de variabilidad mínima requerida
- Control de consistencia interna
- Detección anticipada de un evento de helada.
- Emisión de alertas mediante mensajes SMS
- Publicación de la información climatológica de un evento de helada en el Geoportal

Fig. 27 Ejecución de las pruebas funcionales

```
===== test session starts =====
platform linux -- Python 3.6.8, pytest-4.6.3, py-1.8.0, pluggy-
0.12.0
rootdir: /home/developer/Workspace/frostdetector/tests/func
collected 12 items

test_controls.py ... [ 25%]
test_frost_detector.py ... [ 50%]
test_messaging.py ... [ 75%]
test_ftp.py ... [100%]

===== 12 passed in 3.3 seconds =====
```

Elaborado por: El Investigador

Pruebas de aceptación

Las pruebas de aceptación sirven para comprobar si el software cumple con los requisitos de funcionamiento esperado por el cliente. Para el proyecto, las pruebas fueron definidas por los técnicos de la GIZ y se verificó las siguientes funcionalidades.

- Incorporación de la funcionalidad de detección anticipada una helada y emisión de alertas a una estación meteorológica.
- Detección anticipada de un evento de helada con datos climatológicos válidos.
- Emisión de alertas de acuerdo con los parámetros establecidos en la ficha de la estación.
- Envío mensajes SMS en tiempo real.
- Visualización del historial de las alertas emitidas en el sitio web.

Finalmente, ejecutadas las pruebas satisfactoriamente se procedió a desplegar el proyecto a producción. Los pasos realizados se presentan en el Anexo 2.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- El sistema web para el control, validación y publicación de datos de la red hidrometeorológica de Tungurahua presenta fallos en el proceso de validación de los controles de consistencia máxima permitida y mínima requerida de datos instantáneos dado que no se tomó en cuenta datos faltantes ni los parámetros meteorológicos que deben realizar los controles. Además, se identificó que la duración del procesamiento de la información climatológica es demasiada larga (una hora en validar 60 datos de un minuto).
- En la aparición de un fenómeno meteorológico de helada se determinó que la caída drástica de temperatura por debajo de los 3 °C repercute en la formación de este evento extremo.
- La aplicación de los controles de calidad definidos por la Organización Meteorológica Mundial sobre los datos meteorológicos obtenidos de las estaciones meteorológicas automáticas permitió descartar datos erróneos en el monitoreo de la temperatura, humedad relativa y punto de rocío.
- La emisión de alertas mediante mensajes SMS sobre la posible aparición de un evento de helada ayudó a los agricultores de la zona de Yatzaputzan a reducir las pérdidas de los cultivos dado que, al momento de recibir la alerta de inicio de helada, los agricultores activaron el sistema de riego por aspersión. Además, la alerta permitió a los agricultores resguardar los animales en un sitio más cálido con la finalidad de preservar la salud de los animales.

5.2 Recomendaciones

- Se recomienda emplear conexiones más estables en la transmisión de datos de las estaciones meteorológicas con el fin de proveer la disponibilidad e integridad de los datos. En el caso de la estación meteorología Tamboloma se podría integrar en la red SCADA que utilizan las estaciones de Mula Corral y Chiquihurco.
- Se sugiere utilizar más información climática para la elaboración del algoritmo de detección de heladas, además se podría incorporar modelos predictivos que utilicen información agro-climatológica histórica y técnicas de inteligencia artificial con el fin de hacer más preciso la detección de la helada.
- Se recomienda migrar el sistema web a una arquitectura de microservicios ya que facilitaría la incorporación de nuevas funcionalidades tales como pronósticos o alertas frente a fenómenos meteorológicos extremos.

MATERIALES DE REFERENCIA

Referencias Bibliográficas

- [1] C. D. Arcos Guamán, “Diseño e implementación de un sistema electrónico para el monitoreo, control de la helada y riego, en huertos de fresas por medio de redes de sensores inalámbricos con hardware libre.”, Escuela Superior Politécnica de Chimborazo, 2017.
- [2] E. M. Castro Mejía, “Sistema de control electrónico de plagas, calefacción y regadío en las plantaciones de mora ubicadas en la Provincia de Tungurahua, Parroquia Huachi Grande Sector El Arbolito”, Universidad Técnica de Ambato, 2016.
- [3] C. G. Araujo Criollo, “Diseño e implementación de un prototipo electrónico para el monitoreo y señalización de la producción agrícola de quinua a cielo abierto en la hacienda la Cantora”, Universidad de las Fuerzas Armadas ESPE, 2015.
- [4] W. R. Santos Benavides y B. R. Zarabia Sandoval, “Diseño, automatización y supervisión de un sistema de riego localizado de alta frecuencia, protección contra heladas y dosificación de fertilizantes para la inyección que requiere el cultivo de fragaria (frutilla)”, Universidad de las Fuerzas Armadas ESPE, 2008.
- [5] FIA, INIA, Vinos de Chile, DMC y FDF, “HELADAS. Tipos, medidas de prevención y manejos posteriores al daño. Guía de uso del sitio.”, 2016.
- [6] G. C. Nelson et al., “Cambio Climático: El impacto en la agricultura y los costos de adaptación”, 2009.
- [7] El Universo, “Heladas afectan cultivos en Tungurahua”, 2018. [En línea]. Disponible en: <https://www.eluniverso.com/noticias/2018/06/21/nota/6822023/heladas-afectan-cultivos-tungurahua>. [Accedido: 24-abr-2019].
- [8] IDEAM, Atlas climatológico de Colombia. Bogotá, 2005.
- [9] Organización Meteorológica Mundial, Guía de Instrumentos y Métodos de Observación Meteorológicos. 2014.

- [10] Organización de las Naciones Unidas para la Agricultura y la Alimentación, “Definiciones de congelación y de helada”, en Protección contra las heladas: fundamentos, práctica y economía, Roma, 2010, pp. 2-4.
- [11] L. G. Matías Ramírez, Ó. A. Fuentes Mariles, y F. García Jiménez, “Heladas”, México, D. F., 2012.
- [12] A. Fernandez, Python 3 al descubierto, 2.a ed. México D.F.: Alfaomega, RC Libros, 2013.
- [13] The PHP Group, “¿Qué es PHP?”, 2019. [En línea]. Disponible en: <https://www.php.net/manual/es/intro-whatism.php>.
- [14] S. Bradshaw y K. Chodorow, “Introduction”, en MongoDB: The Definitive Guide, 3.a ed., M. Cronin, Ed. O’Reilly Media, 2019, p. 6.
- [15] British Columbia Institute of Technology, “CodeIgniter User Guide”, 2019. [En línea]. Disponible en: https://www.codeigniter.com/user_guide.
- [16] Highsoft, “Highcharts”. [En línea]. Disponible en: <https://www.highcharts.com/blog/products/highcharts>.
- [17] MongoEngine, “MongoEngine User Documentation”, 2019. [En línea]. Disponible en: <http://docs.mongoengine.org/>.
- [18] O. Okken, Python Testing with Pytest: Simple, Rapid, Effective, and Scalable. Pragmatic Bookshelf, 2017.
- [19] C. M. Kozierok, “FILE TRANSFER PROTOCOL (FTP)”, en The TCP/IP Guide, William Pollock, 2005, pp. 1170-1171.
- [20] A. Rodriguez, “Servicios Web de RESTful: Los aspectos básicos”, p. 10, 2015.
- [21] TomTom Telematics BV, “¿Qué es el GPRS?”, 2018. [En línea]. Disponible en: https://es.support.telematics.tomtom.com/app/answers/detail/a_id/4230/kw/gprs.
- [22] S. Harris y F. Maymí, “Software Development Methodologies”, en CISSP All-in-One Exam Guide, 8.a ed., McGraw-Hill Education, 2019, p. 1142.
- [23] J. Canós, P. Letelier, y C. Penadés, "Métodologías Ágiles en el Desarrollo de Software", 2003.

- [24] A. Navarro Cadavid, J. D. Fernández Martínez, y J. Morales Vélez, “Revisión de metodologías ágiles para el desarrollo de software”, 2013.
- [25] J. Palacio, “Scrum Manager I: Las reglas de scrum”, 2014, p. 22.
- [26] A. Peralta, “Metodología Scrum”, Universidad ORT Uruguay, 2003, p. 3.
- [27] Kanbanize, “Kanban: explicación para principiantes”, 2019. [En línea]. Disponible en: <https://kanbanize.com/es/recursos-de-kanban/primeros-pasos/que-es-kanban/>.
- [28] Universidad Miguel Hernández, “Técnicas Ágiles de Desarrollo de Software (2818). Método Kanban”, 2019. [En línea]. Disponible en: <http://umh2818.edu.umh.es/unidades-didacticas/metodologias/metodo-kanban>.
- [29] M. Skarin, "What Is Kanban?", en Real-World Kanban: Do Less, Accomplish More with Lean Thinking, Pragmatic Bookshelf, 2015, pp. 4-7.
- [30] J. A. Santana León, “Sistema web de control, validación y publicación de datos de la red hidrometeorológica de Tungurahua para el Honorable Gobierno Provincial de Tungurahua”, Universidad Técnica de Ambato, 2017.
- [31] Organización Meteorológica Mundial, “Guidelines on Quality Control Procedures for Data from Automatic Weather Stations”, 2004.
- [32] SENSIRION, “SHTxx Application Note Dew-point Calculation”, 2006.

Anexos

Anexo 1: Historias de usuario

Historia de Usuario	
Número: 1	Nombre: Descarga, validación y almacenamiento de los datos meteorológicos.
Usuario: Administrador del Geoportal	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Media
Descripción: El software debe descargar los archivos con la información climatológica registrada en un intervalo de 1 o 5 minutos desde el servidor FTP del HGPT. Posteriormente los validará con los siguientes controles: <ul style="list-style-type: none">- Control de umbrales- Control de consistencia temporal (máxima permitida y mínima requerida)- Control de consistencia interna. Finalmente se almacenarán en la base de datos.	
Observaciones: En el caso de que la estación meteorológica tenga habilitado la detección de heladas, se debe incluir el parámetro meteorológico de punto de rocío.	

Historia de Usuario	
Número: 2	Nombre: Detección de un posible evento de helada.
Usuario: Administrador del Geoportal	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: El software debe monitorear en tiempo real los datos meteorológicos con la finalidad de detectar un posible evento de helada.	
Observaciones: <ul style="list-style-type: none">- El proceso de detección de heladas debe ser implementado en base al estudio realizado por el M.Sc. Patrick Reuter, técnico de la GIZ.- El monitoreo se debe realizar durante un periodo de tiempo determinado.	

Historia de Usuario	
Número: 3	Nombre: Envío de mensajes SMS al inicio o fin de un evento de helada.
Usuario: Suscriptor al SATH	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Baja
Descripción: El software debe enviar un mensaje SMS a los suscriptores del SATH al momento de que los datos meteorológicos cumplan con las condiciones de inicio o fin de una helada.	
Observaciones: - El contenido de los mensajes va a ser proporcionado por los técnicos de la GIZ y deben ser enviados a aprobar a la dirección de sistemas del HGPT.	

Historia de Usuario	
Número: 4	Nombre: Visualización de los registros de eventos de helada.
Usuario: Administrador del Geoportal, Usuario general	
Prioridad en Negocio: Media	Riesgo en Desarrollo: Baja
Descripción: Mostrar en la página principal del Geoportal una sección para las estaciones meteorológicas con detección de heladas. Además, en la ficha de la estación debe haber una sección para mostrar el historial de heladas en forma de tabla. La tabla deberá tener un espacio para mostrar un botón que genere una gráfica de múltiple eje que muestre la información climatológica de la temperatura, humedad relativa y punto de rocío durante un evento de helada.	
Observaciones:	

Historia de Usuario	
Número: 5	Nombre: Interfaz web para configurar los parámetros del detector de heladas
Usuario: Administrador del Geoportal	
Prioridad en Negocio: Media	Riesgo en Desarrollo: Baja
Descripción: Como administrador, la ficha de la estación debe mostrar un formulario para cambiar los umbrales de temperatura y el periodo de tiempo del monitoreo.	
Observaciones: Utilizar componentes visuales interactivos.	

Historia de Usuario	
Número: 6	Nombre: Interfaz web para administrar los grupos suscritos al SATH.
Usuario: Administrador del Geoportal	
Prioridad en Negocio: Media	Riesgo en Desarrollo: Media
Descripción: Como administrador, en la ficha de la estación debe mostrar un formulario para añadir o quitar los grupos suscritos al SATH.	
Observaciones: La información de los grupos debe provenir del proyecto de mensajería. El proyecto de mensajería centraliza la información en la base de datos Oracle. La creación, edición y visualización de las personas suscritas se realizará en la página de administración del servicio de mensajería del HGPT.	

Historia de Usuario	
Número: 7	Nombre: Interfaz web para cambiar el intervalo de registro de datos para las estaciones meteorológicas.
Usuario: Administrador del Geoportal	
Prioridad en Negocio: Baja	Riesgo en Desarrollo: Baja
Descripción:	

En el formulario de edición o creación de una nueva estación meteorológica debe mostrarse un campo para seleccionar el tipo de intervalo de registro que realiza la estación.

Los intervalos de registro admitidos son de 1 minuto y de 5 minutos.

Observaciones:

Anexo 2: Despliegue del proyecto

El software de procesamiento desarrollado en Python se ejecutará en un servidor Ubuntu 16.04 LTS. Para el despliegue del proyecto en producción se realizó los siguientes pasos:

1. Subida del proyecto al servidor de producción excluyendo archivos de pruebas y de documentación.

Primero se comprimió el proyecto excluyendo archivos de pruebas y de documentación.

```
$ tar -czvf frostdetector.tar.gz ~/Workspace/frostdetector/
```

Luego se subió el archivo comprimido al servidor de producción mediante el protocolo SCP.

```
$ scp frostdetector.tar.gz user@host:/opt/
```

Finalmente se descomprimió el archivo *frostdetector.tar.gz* en el servidor de producción.

```
$ tar -xzvf frostdetector.tar.gz
```

2. Creación de un entorno virtual aislado con una versión de Python 3.6 o superior.

La creación del entorno virtual se realizó con *Virtualenv* y se especificó que se ubique en la raíz del proyecto.

```
$ virtualenv -p python3 venv
```

Finalmente se activó el entorno virtual.

```
$ source venv/bin/active
```

3. Instalación de dependencias.

```
$ pip install -r requirements.txt
```

4. Programación de una tarea en cron para la ejecución del programa en un intervalo de un minuto.

Primero se editó el archivo *crontab* con el siguiente comando.

```
$ crontab -e
```

Una vez abierto el archivo se agregó la siguiente línea.

```
* * * * * /opt/frostdetector/venv/bin/python  
/opt/frostdetector/core.py --env PRODUCTION
```

Anexo 3: Certificado de culminación del Proyecto de Investigación



Honorable Gobierno
Provincial de Tungurahua

Tungurahua para el Ecuador
y el mundo

Ambato, 8 de noviembre de 2019

Ingeniera M.Sc.

Pilar Urrutia U.

DECANA

Facultad de Tecnologías de la Información, Telecomunicaciones e Industrial
Presente

Señor Decano:

Por medio del presente, en calidad de directora de Recursos Hídricos y Conservación Ambiental (E) del Honorable Gobierno Provincial de Tungurahua certifico que el trabajo de investigación: Software de control meteorológico para la detección de heladas y emisión de alertas tempranas en el Geoportal del Honorable Gobierno Provincial de Tungurahua desarrollado por el señor: Paul Alberto Panata Buñay, ha sido concluido de conformidad a los intereses del Honorable Gobierno Provincial de Tungurahua.

Atentamente:



Ing. Olga Paredes

Directora de Recursos Hídricos y Conservación Ambiental (E)