

UNIVERSIDAD ESTATAL DE BOLÍVAR

FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA

CARRERA DE SOFTWARE

TRABAJO DE INTEGRACIÓN CURRICULAR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIEROS EN SOFTWARE

FORMA: PROYECTO TECNOLÓGICO

TEMA:

TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.

AUTORES:

BRYAN HERNÁN GAIBOR NARANJO
FRANKLIN SEBASTIÁN GARCÍA VILLARES

DIRECTOR:

Dr. Henry Fernando Vallejo Ballesteros

GUARANDA – ECUADOR

2024

TEMA DEL PROYECTO TECNOLÓGICO

TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.

DEDICATORIA

Quiero dedicar este proyecto de integración curricular a:

Mis padres, que siempre creyeron en mí, enseñándome el valor del esfuerzo y la perseverancia, este trabajo es un reflejo de las lecciones y valores que me han enseñado a lo largo de los años.

Con todo mi cariño a mi familia, quienes han sido mi refugio y mi apoyo en cada etapa de este camino. Su amor constante, sus palabras de ánimo y su fe en mí me guiaron en los momentos más difíciles.

A mis amigos, que, con su alegría, apoyo incondicional y paciencia, han hecho que este trayecto sea mucho más llevadero. Estoy agradecido por su compañía constante, que me ha recordado que nunca estoy solo y por celebrar junto a mí cada pequeño éxito.

Finalmente, a mi yo más joven, que aspiraba a alcanzar este logro. Espero que este trabajo sea el inicio de nuevas experiencias y aprendizajes en el futuro, y que cada objetivo logrado sea solo el primer paso hacia algo aún más significativo.

Bryan Hernán Gaibor Naranjo

A mi madre y a mi abuelita, cuya presencia ha sido un faro constante en mi vida. Su fortaleza y sabiduría han guiado cada uno de mis pasos. Les agradezco profundamente por inculcar en mí los valores y principios que me han permitido llegar hasta este punto en mi camino.

A mi familia, que con su apoyo y comprensión han sido el cimiento sobre el cual he construido mis sueños. Cada uno de ustedes ha contribuido de manera invaluable a mi crecimiento personal y profesional. Su confianza en mí ha sido una fuente inagotable de motivación.

A mis amigos, que con su compañía y amistad han enriquecido mi vida. Gracias por estar a mi lado en los momentos de alegría y en los desafíos, por ofrecer siempre una palabra de aliento y por compartir esta aventura conmigo. Su amistad es un tesoro que valoro profundamente.

Este proyecto es un reflejo de todo lo que he aprendido de ustedes. Cada logro alcanzado es fruto de las lecciones y el cariño que me han brindado. Sin su influencia y su apoyo, no habría sido posible llegar hasta aquí. Les dedico este esfuerzo con gratitud y reconocimiento, esperando que se sientan tan orgullosos de mí como yo lo estoy de tenerlos en mi vida.

Franklin Sebastián García Villares

AGRADECIMIENTO

En este momento de culminación, quiero expresar mi más profundo agradecimiento a todas las personas que han sido parte fundamental de este viaje académico.
Primero, a mis padres, cuya dedicación y sacrificio han sido el cimiento sobre el que he construido mis aspiraciones. Su apoyo constante y sus consejos han sido una fuente inagotable de inspiración y fortaleza. Agradezco profundamente su fe en mí y su constante presencia a lo largo de este viaje.

A mi familia, por sus palabras de aliento y comprensión que han aliviado los momentos más desafiantes. Su apoyo ha sido un refugio de consuelo y motivación. Gracias por ser mi base y mi mayor fuente de alegría.

A los docentes de la Universidad Estatal de Bolívar, cuya pasión por la enseñanza y compromiso han sido fundamentales para mi formación académica. Sus enseñanzas y guías han iluminado mi camino y enriquecido mi conocimiento de manera invaluable. Agradezco su dedicación y por crear un entorno educativo tan estimulante.

A mis amigos y compañeros, quienes han estado a mi lado brindándome su apoyo y compañía tanto en los momentos de presión como en los buenos instantes. Su amistad ha añadido alegría y significado a esta experiencia. Gracias por ser mi apoyo constante y por todos los momentos compartidos.

A mi director de tesis y pares académicos de mi trabajo de integración curricular, por su orientación experta y paciencia. Su dirección ha sido esencial para convertir este proyecto en una realidad concreta. Estoy agradecido por su dedicación y por estar siempre dispuestos a ofrecernos su orientación y apoyo.

Finalmente, a todos aquellos que, aunque no mencionados específicamente, han influido en mi desarrollo y éxito en este proyecto. Su apoyo y contribuciones, aunque a veces sutiles, han sido vitales para alcanzar este logro. Gracias por su aliento y por creer en mí. Este trabajo es fruto no solo de mi esfuerzo, sino también del apoyo y la inspiración de todos ustedes. ¡Mi más sincero agradecimiento a cada uno!

Bryan Hernán Gaibor Naranjo

Quiero expresar mi más profundo y sincero agradecimiento a mi madre y a mi abuelita, quienes han sido el pilar fundamental en mi vida. Su amor incondicional, su fortaleza y sus enseñanzas han guiado cada uno de mis pasos. Mamá, tu dedicación y sacrificio han sido una inspiración constante; me has enseñado el valor del esfuerzo y la perseverancia. Abuelita, tus palabras llenas de sabiduría y tu cálido abrazo han sido mi refugio en los momentos más difíciles. Sin su apoyo y su fe inquebrantable en mí, este logro no habría sido posible.

A mi familia, gracias por ser el cimiento que sostiene mis sueños. Cada uno de ustedes ha contribuido con su cariño y comprensión a que hoy pueda alcanzar esta meta. Su confianza en mí ha sido la luz que ha iluminado mi camino, incluso en las etapas más desafiantes.

A mis amigos, compañeros de esta travesía, quiero agradecerles por su amistad sincera y por los momentos compartidos que han enriquecido mi vida. Sus risas, consejos y compañía han sido esenciales para mantenerme motivado y recordar que el camino es más llevadero cuando se recorre en buena compañía.

Quiero expresar mi más profundo agradecimiento al director de este proyecto, Dr. Henry Vallejo, por su guía, paciencia y dedicación. Su experiencia y conocimientos han sido una fuente invaluable de aprendizaje, y su confianza en mis capacidades me ha motivado a superar mis propios límites.

A los pares académicos y profesores que han formado parte de este proceso, gracias por compartir su sabiduría y por sus valiosos comentarios y críticas constructivas. Sus enseñanzas han sido fundamentales para mi crecimiento profesional y personal.

Finalmente, a todas las personas que, de una u otra forma, han estado presentes en este camino, les doy las gracias de todo corazón. Este proyecto es el reflejo de cada uno de ustedes, y es un honor compartir este logro con quienes han dejado una huella imborrable en mi vida.

Franklin Sebastián García Villares

CERTIFICADO DE VALIDACIÓN

DR. HENRY VALLEJO, ING.GALUTH GARCÍA y ING. DANILO BARRENO, en su orden Director y Pares Académicos del Trabajo de Integración Curricular "TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.", desarrollado por los estudiantes BRYAN HERNAN GAIBOR NARANJO y FRANKLIN SEBASTIAN GARCIA VILLARES.

CERTIFICAN

Que, luego de revisado el Trabajo de Integración Curricular en su totalidad, cumple con las exigencias académicas de la carrera SOFTWARE, por lo tanto, autorizamos su presentación y defensa.

Guaranda, 09 de diciembre del 2024

Firmado electrónicamente por:
HENRY FERNANDO VALLEJO BALLESTEROS

DR. HENRY VALLEJO

Director

Firmado electrónicamente por:
DANILO GEOVANNY BARRENO NARANJO

ING.DANILO BARRENO

Par Académico

Firmado electrónicamente por:
GALUTH IRENE GARCIA CAMACHO

ING. GALUTH GARCÍA

Par Académico

DERECHOS DE AUTOR

Nosotros **Bryan Hernán Gaibor Naranjo y Franklin Sebastián García Villares** portadores de la cédula de identidad **N°1752326007 y 0202077640** respectivamente**,** en calidad de autores y titulares de los derechos morales y patrimoniales del Trabajo de Titulación: "TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR "**,** modalidad Trabajo de Integración Curricular, de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, concedemos a favor de la Universidad Estatal de Bolívar, una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservamos a nuestro favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizamos a la Universidad Estatal de Bolívar para que realice la digitalización y publicación de este trabajo de titulación el Repositorio Digital, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Los autores declaran que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Bryan Hernán Gaibor Naranjo
CI. 1752326007

Franklin Sebastián García Villares
CI. 0202077640

INDICE DE CONTENIDOS

INTRODUCCIÓN **1**

RESUMEN **2**

ABSTRACT **3**

1 CAPÍTULO I FORMULACIÓN GENERAL DEL PROYECTO **4**

1.1 TEMA 4
1.2 DESCRIPCIÓN DEL PROBLEMA 4
1.3 JUSTIFICACIÓN 5
1.4 OBJETIVOS: 6

2 CAPÍTULO II MARCO TEÓRICO **7**

2.1 ANTECEDENTES 7
2.2 CIENTÍFICO 8
2.3 CONCEPTUAL 12
2.4 LEGAL 16
2.5 GEORREFERENCIAL 19

3 CAPITULO III METODOLOGÍA **20**

3.1 METODOLOGÍA DE DESARROLLO DE SOFTWARE 20
3.2 TÉCNICAS E INSTRUMENTOS DE RECOPILACIÓN DE DATOS 25

4 CAPITULO IV INGENIERÍA DEL PROYECTO **26**

4.1 ANÁLISIS 26
4.1.1 Especificación de Requerimientos de Software 26
4.1.1.1 Introducción 26
4.1.1.2 Alcance del Producto 26
4.1.1.3 Definiciones, Acrónicos y Abreviaturas 26
4.1.1.4 Referencias 28
4.1.2 Descripción General del Producto 28
4.1.2.1 Perspectiva del producto 28
4.1.2.2 Funciones del producto 29
4.1.2.3 Características de los usuarios 30

4.1.2.4 Restricciones generales 30
4.1.3 Requerimientos de Usuario 31
4.1.3.1 Requerimientos funcionales 31
4.1.3.2 Requerimientos no funcionales 32
4.1.3.3 Requerimientos de interfaz 35
4.2 DISEÑO 37
4.2.1 Arquitectura del Software 37
4.2.2 Diagramas 40
4.2.2.1 Diagrama de Caso de Uso 40
4.2.2.2 Diagrama de Secuencia 41
4.2.3 Diseño de la Interfaz 42
4.2.3.1 Patrón de navegación 42
4.2.3.2 Interfaz de Pantallas 43
4.2.3.3 Interfaz de visor Grintec 44
4.3 PROGRAMACIÓN 45
4.3.1 Definición del objetivo 45
4.3.2 Análisis del problema 45
4.3.3 Diseño del Algoritmo 46
4.4 CODIFICACIÓN 48
Procesamiento de archivos (enviar.py) 50
Descripción de los Archivos 53
4.5 PRUEBAS 54
4.5.1 Análisis de requisitos 54
4.5.2 Planificación de pruebas 54
4.5.3 Diseño y desarrollo de casos de prueba 55
4.5.4 Configuración del entorno de prueba 56
4.5.5 Ejecución de la prueba 57
4.5.6 Prueba de cierre 62
5 CONCLUSIONES 63
6 RECOMENDACIONES. 64
7 BIBLIOGRAFÍA 65
8 ANEXOS 68

INDICE DE TABLAS

Tabla 1.Canales GOES-16 11
Tabla 2 Definiciones de acrónimos, siglas y abreviaturas usadas en el documento ERS 26
Tabla 3 Referencias 28
Tabla 4 Tipo de Usuario - Usuario 30
Tabla 5 Tipo de Usuario - Administrador 30
Tabla 6 Requerimiento funcional 01: Selección de Canal y Fecha 31
Tabla 7 Requerimiento funcional 02: Visualización de animaciones 31
Tabla 8 Requerimientos no funcionales 01: Funcionalidad 32
Tabla 9 Requerimientos no funcionales 02: Usabilidad 32
Tabla 10 Requerimientos no funcionales 03: Disponibilidad 32
Tabla 11 Requerimientos no funcionales 04: Compatibilidad 33
Tabla 12 Requerimientos no funcionales 05: Rendimiento 33
Tabla 13 Requerimientos no funcionales 06: Seguridad 33
Tabla 14 Requerimientos no funcionales 07: Generación Automática de Animaciones 34
Tabla 15 Requerimientos no funcionales 08: Carga Automática al Servidor. 34
Tabla 16 Requerimientos no funcionales 09: Automatización del recorte y envió de archivos 35
Tabla 17 Requerimiento de interfaz 01: Interfaz de usuario 35
Tabla 18 Requerimiento de interfaz 02: Compatibilidad de navegadores 36
Tabla 19 Requerimiento de interfaz 03: Interfaz de hardware 36
Tabla 20 Requerimiento de interfaz 04: Interfaz de software 36
Tabla 21 Análisis de requisitos 54
Tabla 22 Cronograma de pruebas 55
Tabla 23 Caso de Prueba 01: Selección de Canal y Fecha 55
Tabla 24 Caso de Prueba 02: Visualización de Animaciones 56
Tabla 25 Ficha de prueba de funcionalidad 57
Tabla 26 Ficha - Evaluación de Rendimiento 58
Tabla 27 Ficha - Evaluación de Usabilidad 59
Tabla 28 Pruebas de Caja Negra 60
Tabla 29 Pruebas de Caja Blanca 61
Tabla 30 Prueba de Cierre para el Proyecto "Visor Grintec" 62

INDICE DE FIGURAS

Figura 1. Imagen Pancromática, producto en niveles de grises..... 9

Figura 2. Diferencia entre imagen RGB, Multiespectral e Hiperespectral..... 10

Figura 3. Funcionamiento FTP..... 13

Figura 4. Mapa de temperatura de la Provincia Bolívar..... 19

Figura 5. Arquitectura cliente-servidor 39

Figura 6. Diagrama de Casos de Uso 40

Figura 7 Diagrama de secuencia..... 41

Figura 8. Patrón de navegación del usuario..... 42

Figura 9. Paleta de colores..... 43

Figura 10. Tipografía Noto Sans 44

Figura 11. Interfaz del visor Grintec 44

Figura 13. Librerías utilizadas en script recorte.py 48

Figura 14. Librerías utilizadas en el script enviar.py 51

Figura 15 Estructura del código..... 53

INTRODUCCIÓN

El proyecto tecnológico titulado “Time lapse a partir de imágenes satelitales en tiempo real del GOES-16, en la provincia Bolívar.” tiene como resultado final una aplicación web para la visualización de animaciones climáticas. Su propósito es ofrecer una herramienta visual que permita monitorear las condiciones meteorológicas en la provincia de Bolívar, utilizando datos en tiempo real proporcionados por el satélite GOES-16. Actualmente, en la provincia de Bolívar no existen herramientas que presenten información climática en tiempo real de manera visual y accesible para el público general.

La metodología adoptada para este proyecto se basa en el enfoque ágil Scrum, que promueve un desarrollo iterativo facilitando la colaboración del equipo y la rápida adaptación a cambios. Scrum se organiza en ciclos cortos y revisiones periódicas, lo que posibilita la mejora continua del producto. En este contexto, cada ciclo incluye el desarrollo y la optimización de técnicas de procesamiento de imágenes que automatizan la generación de animaciones time lapse a partir de los datos satelitales. Estas animaciones, actualizadas de forma periódica sin necesidad de intervención manual, mejoran la comunicación y difusión de la información climática al ofrecer una representación visual dinámica del clima en la provincia de Bolívar. La meta de la aplicación web es incrementar la disponibilidad y accesibilidad de los datos climáticos en tiempo real, impulsando avances tecnológicos y facilitando el acceso a información relevante para los residentes de la región.

La documentación relacionada con el desarrollo de la aplicación web para la visualización de animaciones climáticas está organizada de la siguiente manera: En el capítulo I se presenta la formulación general del proyecto, incluyendo la descripción del problema, la justificación y los objetivos. El capítulo II aborda el marco teórico, los antecedentes, los conceptos clave, el marco legal y la georreferenciación. En el capítulo III se detalla la metodología del proyecto, abarcando tanto el enfoque de desarrollo como el método de investigación utilizado para la recopilación de información. Finalmente, el capítulo IV se enfoca en la ingeniería del proyecto, describiendo cada fase del desarrollo conforme a las técnicas de procesamiento y automatización implementadas.

RESUMEN

El proyecto tecnológico denominado “Time Lapse a partir de imágenes satelitales en tiempo real del GOES-16 en la provincia de Bolívar” ha culminado exitosamente con el desarrollo de una aplicación web llamada “Visor Grintec”. Esta herramienta está diseñada para proporcionar una visualización avanzada de las condiciones climáticas en la provincia de Bolívar mediante animaciones creadas a partir de datos satelitales en tiempo real del satélite GOES-16, accesibles a través del portal Grintec.org. Hasta la fecha, en la región no existían soluciones que ofrecieran información climática en tiempo real de manera visual y fácilmente accesible para el público general.

Para llevar a cabo este proyecto se utilizó la metodología ágil Scrum, la cual se caracteriza por ciclos de trabajo cortos y revisiones periódicas que facilitan la realización de ajustes y mejoras continuas. Dentro de este marco, se implementaron técnicas avanzadas de procesamiento de imágenes y se automatizaron tareas programadas, garantizando un procesamiento autónomo y eficiente de las imágenes satelitales. Esto permitió la generación de animaciones que se actualizan automáticamente sin necesidad de intervención humana, mejorando así la presentación y difusión de la información climática de manera dinámica y precisa en la provincia de Bolívar.

El Visor Grintec está diseñado para aumentar significativamente la disponibilidad y accesibilidad de los datos climáticos, promoviendo el desarrollo tecnológico en la región y facilitando el acceso a información en tiempo real para los residentes de la provincia de Bolívar. La aplicación responde de manera efectiva a la necesidad de contar con una herramienta visual moderna para el monitoreo del clima, enfocándose en la facilidad de uso, la precisión de los datos y la eficiencia en la actualización de la información.

Palabras claves: Time Lapse GOES-16, imágenes satelitales, monitoreo climático, Bandas Espectrales.

ABSTRACT

The technological initiative titled “Time Lapse from Real-Time GOES-16 Satellite Images in Bolívar Province” has successfully culminated in the development of a web application named “Visor Grintec.” This application is crafted to deliver an advanced visualization of weather conditions in Bolívar Province through animations generated from real-time satellite data provided by the GOES-16 satellite, accessible via the Grintec.org portal. Prior to this project, the region lacked tools that offered real-time climatic information in a visually engaging and easily accessible format for the general public.

To execute this project, the agile Scrum methodology was employed, characterized by short work cycles and regular reviews that facilitate continuous adjustments and improvements. Within this framework, advanced image processing techniques were implemented, and scheduled tasks were automated to ensure autonomous and efficient handling of satellite images. This approach enabled the creation of animations that update automatically without human intervention, thereby enhancing the dynamic and accurate presentation and dissemination of climatic information in Bolívar Province.

Visor Grintec is designed to significantly improve the availability and accessibility of climatic data, fostering technological advancement in the region and providing residents of Bolívar Province with real-time information. The application effectively addresses the need for a modern visual tool for climate monitoring, emphasizing user-friendliness, data accuracy, and efficient information updates.

Keywords: Time Lapse GOES-16, satellite images, climate monitoring. Spectral Bands

1 CAPÍTULO I FORMULACIÓN GENERAL DEL PROYECTO

1.1 Tema

Time lapse a partir de imágenes satelitales en tiempo real del GOES-16, en la Provincia Bolívar.

1.2 Descripción del Problema

La situación actual se caracteriza por la limitada accesibilidad y la presentación estática de las imágenes satelitales del GOES-16, uno de los satélites más avanzados en términos de observación meteorológica en tiempo real. Estas imágenes, aunque ricas en información, se presentan como instantáneas estáticas, lo que limita significativamente la capacidad de los usuarios para comprender con precisión los patrones climáticos y su evolución en el tiempo. La falta de dinamismo en la presentación de datos climáticos obstaculiza la capacidad de los meteorólogos, investigadores, y el público en general para realizar un seguimiento adecuado de los eventos meteorológicos en curso, como tormentas, frentes climáticos, y otros fenómenos atmosféricos. Las imágenes estáticas no solo carecen de la capacidad de representar el movimiento y la evolución de estos fenómenos, sino que también suelen tener una cobertura geográfica amplia, lo que dificulta el enfoque en áreas específicas de interés, de la provincia Bolívar.

Esta deficiencia en la presentación dinámica de los datos meteorológicos impide una interpretación completa y oportuna de los mismos, lo que a su vez dificulta la toma de decisiones informadas en diversos sectores. Por ejemplo, en el ámbito de la gestión de desastres naturales, la agricultura, la planificación urbana y otras áreas sensibles a las condiciones climáticas, la capacidad de anticipar y responder adecuadamente a los eventos meteorológicos es crucial para minimizar riesgos y maximizar la resiliencia.

1.3 Justificación

La presentación estática de imágenes satelitales del GOES-16 es una limitación significativa en la comprensión precisa de los patrones climáticos en tiempo real. Esta falta de dinamismo dificulta la toma de decisiones informadas por parte de los usuarios en diversos ámbitos, desde la gestión de desastres naturales hasta la planificación urbana y la agricultura.

La aplicación informática propuesta busca abordar esta problemática al ofrecer animaciones en tiempo real generadas a partir de imágenes satelitales del GOES-16. Esta solución permitirá una visualización dinámica y detallada de la evolución de los fenómenos meteorológicos, como tormentas, sistemas frontales y otros eventos climáticos relevantes.

Los beneficios esperados de esta aplicación son significativos. Por un lado, facilitará a las autoridades y organismos de respuesta anticiparse y prepararse mejor para eventos climáticos extremos, minimizando así los riesgos y maximizando la resiliencia frente a desastres naturales.

Además, la aplicación beneficiará a sectores como la agricultura, permitiendo a los agricultores monitorear las condiciones climáticas y planificar actividades agrícolas de manera más eficiente para optimizar los rendimientos y minimizar las pérdidas.

La disponibilidad de estas animaciones en un sitio web accesible, como https://www.grintec.org, garantizará su acceso a una amplia gama de usuarios, promoviendo una mayor conciencia y preparación frente a eventos climáticos adversos.

En el contexto de la carrera de Software, este proyecto contribuye al avance en la línea de investigación de Geomática, específicamente en la sublínea de Gestión, análisis y tratamiento de información espacial. Al utilizar tecnologías y herramientas de vanguardia en el desarrollo de software, se amplía el conocimiento y la aplicación de técnicas de procesamiento de imágenes satelitales y análisis climático, impulsando así futuras investigaciones en este campo.

1.4 Objetivos:

General

Generar animaciones (Time Lapse) a partir de imágenes satelitales en tiempo real del GOES-16, en la provincia de Bolívar.

Específicos

- Desarrollar la aplicación informática que genera el time lapse a partir de las imágenes satelitales en tiempo real del satélite GOES-16, evitando la pérdida de información.
- Crear la aplicación informática que permita subir las imágenes al servidor grintec.org.
- Implementar el módulo web que permita la visualización de las animaciones generadas mediante la aplicación informática.

2 CAPÍTULO II MARCO TEÓRICO

2.1 Antecedentes

En las últimas décadas, se ha evidenciado un crecimiento exponencial en el uso de animaciones como herramienta visual clave en diversas áreas, incluida la educación, la meteorología y la comunicación.

Según (Christian Andres Dávila Vilca, 2021) en su Trabajo de Suficiencia Profesional: "Actualización E Implementación Del Sistema De Recepción Satelital Goes-16 Para El Pronóstico Del Tiempo Atmosférico", realizado para la Universidad Nacional Agraria La Molina, nos dice que los datos y productos obtenidos a través de satélites son fundamentales para las actividades de los servicios meteorológicos nacionales. Se utilizan en las operaciones internas para pronósticos y alertas meteorológicas de mesoescala y a corto plazo. Es crucial mantener actualizados los métodos de adquisición y procesamiento de datos para seguir el ritmo de los avances tecnológicos en la observación satelital y adaptarse a los cambios emergentes.

Para (Orlando H. Álvarez Hernández et al., 2018) en su trabajo "HOMOGENEIZACIÓN DE SERIES DE RAPIDEZ VIENTO MENSUALES EN LAS ESTACIONES METEOROLÓGICAS DEL INAMHI, EN LOJA, ECUADOR.", en el campo de la meteorología y la visualización de datos, un problema que se afronta en la práctica al trabajar con series temporales de observaciones es que es muy común encontrar defectos como la existencia de lagunas (falta de datos) y/o falta de homogeneidad (derivadas de cambios en los parámetros de observación, o por errores en las mismas). Las series relacionadas con la rapidez y la dirección del viento son esenciales en problemas de generación de energía eólica, agrícola, medio ambiental, etc.

En el Trabajo de Suficiencia Profesional "ACTUALIZACIÓN E IMPLEMENTACIÓN DEL SISTEMA DE RECEPCIÓN SATELITAL GOES-16 PARA EL PRONÓSTICO DEL TIEMPO ATMOSFÉRICO", realizado por (Christian Andres Dávila Vilca, 2021) nos explica que, desde 1975, los Satélites Ambientales Operacionales Geoestacionarios (GOES) han proporcionado una amplia gama de imágenes y datos sobre las condiciones atmosféricas y la actividad solar. Estos datos y productos satelitales han

permitido pronósticos meteorológicos más precisos y oportunos, además de mejorar la comprensión del clima a largo plazo. Las imágenes satelitales se utilizan principalmente para localizar y seguir fenómenos meteorológicos severos, identificar áreas con riesgo de incendios o incendios activos, analizar la probabilidad de inundaciones o zonas ya inundadas, evaluar cosechas, y monitorear la contaminación del aire o del agua. Además, sirven para la investigación en diversas disciplinas relacionadas con estos temas y otros similares.

2.2 Científico

Ingeniería de software

La ingeniería de software se refiere a la aplicación de principios y prácticas de ingeniería para el desarrollo, mantenimiento y evolución de sistemas de software de alta calidad. Implica el uso de metodologías, herramientas y técnicas para gestionar de manera efectiva los recursos, los plazos y los riesgos en el proceso de desarrollo de software. (Ian Sommerville, 2015)

Metodologías Agiles (SCRUM)

Las metodologías ágiles no son simplemente un conjunto de directrices específicas para el desarrollo de software. Más bien, representan una mentalidad centrada en la colaboración y la eficiencia del trabajo, estableciendo valores fundamentales que orientan nuestras decisiones sobre qué hacer y cómo hacerlo. Entre las diversas metodologías ágiles se incluyen enfoques como Scrum, Kanban y programación extrema, que ofrecen estructuras flexibles para la gestión de proyectos. (Hat Red, 2022)

Según (Cesar et al., 2018) para la implementación de una metodología de desarrollo de software en aplicaciones geográficas, se requiere entender el grado de especialización de la información para alcanzar resultados satisfactorios en cuanto a las necesidades de los geousuarios. Por lo que, Scrum es señalada como una metodología de desarrollo de software ágil, lo cual permite alcanzar los objetivos planteados.

Satélites meteorológicos

Antes de la implementación de la tecnología satelital para fines meteorológicos, las observaciones se realizaban desde la superficie o mediante aeronaves. Esto significaba que no había un método eficaz para monitorear los

sistemas nubosos durante el desarrollo de fenómenos atmosféricos que pudieran afectar el bienestar social o productivo. (CHRISTIAN ANDRES DÁVILA VILCA, 2021)

Imágenes pancromáticas

Una imagen pancromática utiliza una única banda que combina las bandas roja, verde y azul, lo que permite una mayor resolución espacial. La imagen resultante no contiene información específica sobre la longitud de onda. Una banda pancromática se puede emplear para mejorar la resolución espacial de las imágenes multiespectrales mediante el método de refinamiento pancromático. Este método consiste en combinar las bandas de color de baja resolución con la banda de alta resolución en escala de grises (PAN) para generar una imagen en color de alta resolución. Un posible inconveniente de este método es la distorsión espectral, que puede complicar la clasificación de los objetos. (*Banda Pancromática Y El Uso En Las Imágenes Satelitales*, n.d.)

Figura 1. Imagen Pancromática, producto en niveles de grises

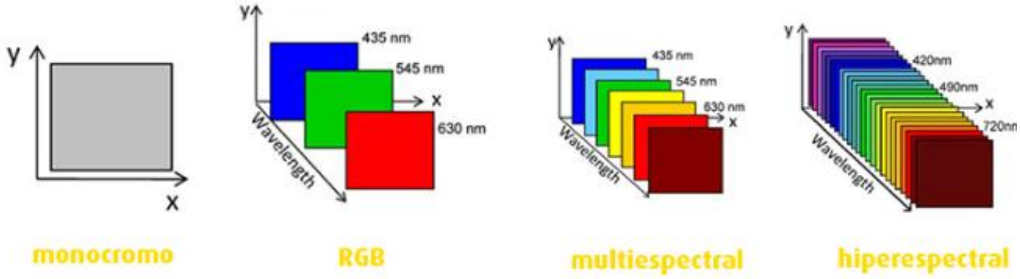


Fuente. (*Banda Pancromática Y El Uso En Las Imágenes Satelitales*, n.d.)

Imágenes multiespectrales

Una imagen multiespectral captura datos de la imagen en rangos específicos de longitudes de onda a lo largo del espectro electromagnético. Estas imágenes dividen la luz en un pequeño número de bandas espectrales, generalmente entre 3 y 15 bandas por imagen. Cuando una captura abarca cientos de bandas espectrales contiguas, se denomina "imagen hiperespectral". (Christian Andres Dávila Vilca, 2021)

Figura 2. Diferencia entre imagen RGB, Multiespectral e Hiperespectral



Fuente. (*Cámaras Hiperespectrales*, n.d.)

GOES-16

La serie de satélites ambientales operacionales geoestacionarios (GOES)-R (GOES R, S, T y U) es el bloque de satélites geoestacionarios de reemplazo que proporciona continuidad para la próxima generación de usuarios y ofrece la primera actualización significativa en la capacidad de los instrumentos en décadas. Se desarrollaron seis instrumentos nuevos y mejorados para observar la Tierra, el Sol y el entorno meteorológico espacial para cada uno de los cuatro satélites de la serie GOES-R, así como un nuevo sistema terrestre para la generación, el acceso y la distribución de productos. (Danielle Losos, 2021)

El sensor ABI posee un conjunto ampliado de 16 canales espectrales en las regiones visible, infrarrojo cercano e infrarrojo del espectro electromagnético, mientras que la generación pasada solo presentaba canales en las regiones visible e infrarrojo.

Tabla 1.Canales GOES-16

ABI (Canal)	Longitud de onda centrada (µm)	Tipo	Nombre	Resolución espacial (km)
01	0.47	Visible	Blue	1
02	0.64	Visible	Red	0.5
03	0.86	Infrarrojo Cercano	Vegetal	1
04	1.37	Infrarrojo Cercano	Cirrus	2
05	1.6	Infrarrojo Cercano	Nieve/Hielo	1
06	2.2	Infrarrojo Cercano	Tamaño de Partículas	2
07	3.9	Infrarrojo	Ventana de Onda Corta	2
08	6.2	Infrarrojo	Vapor de Agua en Nivel Alto	2
09	6.9	Infrarrojo	Vapor de Agua en Nivel Medio	2
10	7.3	Infrarrojo	Vapor de Agua en Nivel Bajo	2
11	8.4	Infrarrojo	Tope de fase de nube	2
12	9.6	Infrarrojo	Ozono	2
13	10.3	Infrarrojo	Ventana “Limpia”	2
14	11.2	Infrarrojo	Ventana Onda Larga	2
15	12.3	Infrarrojo	Ventana “Sucia”	2
16	13.3	Infrarrojo	CO2	2

Nota. En la presente tabla se muestran los canales que conforman al generador avanzado de imágenes (ABI) del satélite GOES-16, listándose los canales y sus respectivas longitudes de onda centradas en el rango del espectro electromagnético que les corresponde, el tipo de banda, nombre de banda y su resolución espacial. (Schmit et al., 2018)

2.3 Conceptual

Cliente-Servidor

El modelo cliente-servidor, también denominado “principio cliente-servidor”, es un esquema de comunicación que facilita la distribución de tareas en una red de computadoras. Un servidor puede ser un dispositivo físico que proporciona recursos a otros sistemas o aplicaciones, o puede ser un software que interactúa con los clientes. Este servidor recibe solicitudes del cliente, las procesa y envía las respuestas correspondientes. Por otro lado, existen diversos tipos de clientes. Un ordenador o una aplicación se comunica con el servidor, enviando solicitudes y recibiendo respuestas. En esencia, el modelo cliente-servidor ilustra la interacción entre un servidor y sus clientes. (*¿Qué Es El Modelo Cliente-Servidor? Pros y Contras - IONOS MX,* n.d.)

Debian

El Proyecto Debian es una comunidad de personas unidas con el objetivo de crear un sistema operativo (SO) libre, conocido como Debian.

Un sistema operativo consiste en un conjunto de programas y herramientas básicas que permiten que una computadora funcione. El componente central de un sistema operativo es el núcleo (kernel), que es el programa más importante en la computadora, encargándose de las tareas fundamentales y permitiendo la ejecución de otros programas.

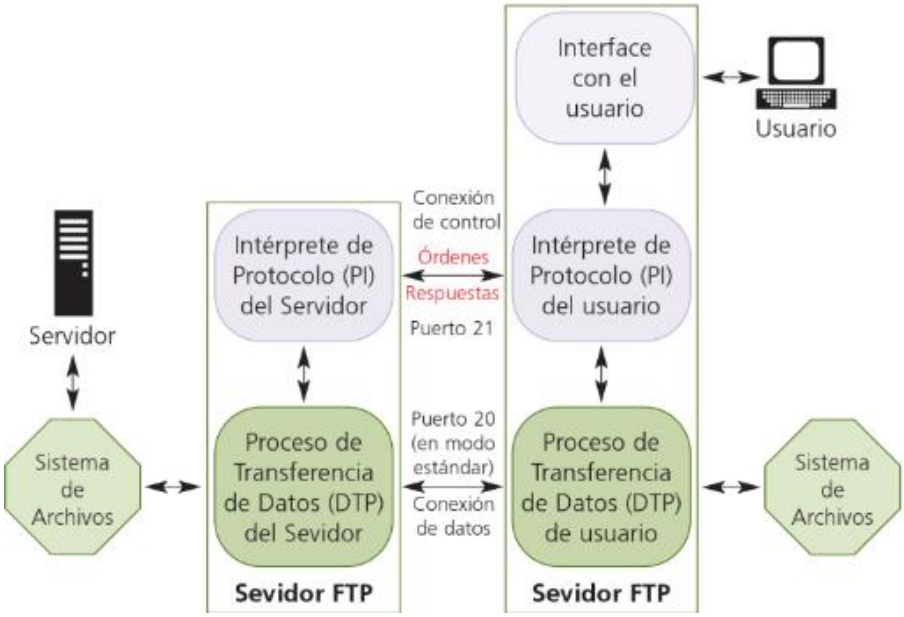
Actualmente, los sistemas Debian utilizan el núcleo de Linux o de FreeBSD. Linux, inicialmente desarrollado por Linus Torvalds, ha sido mejorado por miles de programadores en todo el mundo.

Además, se están realizando esfuerzos para ofrecer Debian con otros núcleos, particularmente con el Hurd. El Hurd es una colección de servidores que operan sobre un micronúcleo (como Mach) para proporcionar diversas funcionalidades. (Debian, 2023)

FTP

El servicio FTP se basa en el protocolo FTP, acrónimo de File Transfer Protocol (Protocolo de Transferencia de Archivos) en redes tipo TCP/IP. Este nombre puede llevar a confusión, ya que no se trata únicamente de un servicio de transferencia de archivos bidireccional entre cliente y servidor. En realidad, es un servicio de administración de archivos que permite realizar diversas acciones, como subir, bajar, borrar, renombrar y mover archivos, así como crear y eliminar carpetas. Estas operaciones se pueden realizar independientemente de los diferentes sistemas de archivos que se utilicen (como FAT16 en DOS, FAT32 y NTFS en Windows, o Ext3 en Linux). (*Servicios FTP (Servicios En Red) - Joaquín Andreu - Google Libros*, n.d.)

Figura 3. Funcionamiento FTP



Fuente. (*Servicios FTP (Servicios En Red) - Joaquín Andreu - Google Libros*, n.d.)

Python

Python es un lenguaje de programación muy popular en aplicaciones web, desarrollo de software, ciencia de datos y machine learning (ML). Los desarrolladores prefieren Python debido a su eficiencia y facilidad de aprendizaje, así como por su compatibilidad con diversas plataformas. Además, Python es de descarga gratuita, se integra fácilmente con distintos sistemas y acelera el proceso de desarrollo. (Charles R. Severance, 2020)

Crontab

Crontab es una herramienta en sistemas operativos Unix y Linux que permite programar la ejecución automática de scripts o comandos en intervalos de tiempo específicos. Las tareas programadas, conocidas como "jobs", se ejecutan según las especificaciones definidas en el archivo crontab. (*Crontab(5) - Linux Manual Page*, n.d.)

PM2

PM2 es un administrador de procesos avanzado para aplicaciones Node.js que permite mantener las aplicaciones en ejecución en segundo plano y facilita su monitoreo, reinicio y balanceo de carga. Su principal ventaja es garantizar la continuidad de las aplicaciones Node.js en entornos de producción, incluso si el servidor se reinicia o si la aplicación falla. (*PM2 - Overview | PM2 Plus Documentation*, n.d.)

Node.js

Node.js es un entorno de ejecución para JavaScript basado en eventos, diseñado para construir aplicaciones escalables y de alto rendimiento. Utiliza un modelo de E/S no bloqueante y dirigido por eventos, lo que permite manejar múltiples conexiones simultáneamente de manera eficiente.

En este proyecto, Node.js se utiliza para ejecutar el servidor backend que gestiona las solicitudes HTTP y sirve la interfaz web para la visualización de videos generados a partir de imágenes satelitales. La elección de Node.js proporciona un entorno rápido y eficiente para desarrollar aplicaciones web interactivas en tiempo real.(*Node.Js — About Node.Js®*, n.d.)

Express.js

Express.js es un framework minimalista para Node.js que facilita la creación de aplicaciones web y API. Ofrece una serie de herramientas y características

que permiten desarrollar aplicaciones robustas de forma rápida, como enrutamiento de peticiones HTTP, manejo de middleware y gestión de sesiones.

Raspberry Pi

La Raspberry Pi es una serie de ordenadores monoplaca u ordenadores de placa simple de bajo costo desarrollado en el Reino Unido por la Raspberry Pi Foundation, con el objetivo de poner en manos de las personas de todo el mundo el poder de la informática y la creación digital. (*Raspberry Pi Documentation,* n.d.)

Sensores Remotos

Los sensores remotos son sistemas o instrumentos diseñados para captar información de objetos a distancia. La teledetección, o percepción remota (remote sensing), se refiere tanto a la adquisición de datos de la superficie terrestre mediante un sensor remoto como al procesamiento e interpretación de esos datos. Las imágenes obtenidas desde satélites son muy valiosas para la cartografía geológica, ya que ofrecen una visión sinóptica de grandes áreas bajo condiciones de iluminación homogéneas, lo que es especialmente útil para detectar estructuras y accidentes a escala regional. El carácter multiespectral y digital de la información registrada por los sensores remotos ha facilitado estudios espectrales que permiten discriminar diferentes litologías. Los datos multiespectrales proporcionan información crucial para diferenciar suelos y rocas según su composición mineralógica. (Guzmán-Álvarez et al., 2022)

Time Lapse

El time lapse consiste en una serie de fotografías tomadas a intervalos regulares, que luego se ensamblan mediante la edición de video para crear una secuencia que armoniza velocidad y tiempo. Esta técnica es ideal para mostrar procesos que ocurren en un período específico. Es comúnmente utilizada para capturar la evolución de paisajes naturales, el desarrollo de obras de construcción, o el flujo de tráfico y personas en un área determinada. (Time Lapse: Qué Es y Cómo Hacerlo | Emilio Simón, n.d.)

Visual Studio Code

Visual Studio Code es un editor de código fuente ligero pero potente que se ejecuta en el escritorio y está disponible para Windows, macOS y Linux. Viene con soporte integrado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes y tiempos de ejecución (como C++, C#, Java, Python, PHP, Go, .NET). (Visual Studio Code, n.d.)

2.4 Legal

ISO IEC 20246: Ingeniería de Software

La norma ISO IEC 20246 reemplaza a la IEEE 1028 y proporciona un marco general para la revisión de productos de trabajo. Considera cualquier artefacto producido durante un proceso como un producto de trabajo. Esta norma es aplicable a cualquier organización que gestione, desarrolle, pruebe y mantenga sistemas y software. Incluye un proceso genérico con actividades, tareas, técnicas de revisión y plantillas de documentación aplicables en cualquier fase del ciclo de vida del producto. Está dirigida a gerentes de proyecto, desarrollo, calidad y pruebas, así como a analistas comerciales, desarrolladores, probadores, clientes y todos aquellos involucrados en el desarrollo, pruebas y mantenimiento de sistemas y software. (Carlos Lobo, 2022)

Ley Orgánica de Protección de Datos

Art. 6.-Accesibilidad y confidencialidad. -Son confidenciales los datos de carácter personal. El acceso a estos datos solo será posible cuando quien los requiera se encuentre debidamente legitimado, conforme a los parámetros previstos en la Ley Orgánica de Protección de Datos Personales, su respectivo reglamento y demás normativa emitida por la Autoridad de Protección de Datos Personales.

Art. 38.-Medidas de seguridad en el ámbito del sector público. -El mecanismo gubernamental de seguridad de la información deberá incluir las medidas que deban implementarse en el caso de tratamiento de datos personales para hacer frente a cualquier riesgo, amenaza, vulnerabilidad, accesos no autorizados, pérdidas, alteraciones, destrucción o comunicación accidental o ilícita en el tratamiento de los datos conforme al principio de seguridad de datos personales.

El mecanismo gubernamental de seguridad de la información abarcará y aplicará a todas las instituciones del sector público, contenidas en el artículo 225 de la Constitución de la República de Ecuador, así como a terceros que presten servicios públicos mediante concesión u otras figuras legalmente reconocidas. Estas, podrán incorporar medidas adicionales al mecanismo gubernamental de seguridad de la información. (Asamblea Nacional del Ecuador, 2022)

El Código Orgánico de la Economía Social del Conocimiento, Creatividad e Innovación

Disposiciones especiales sobre ciertas obras

Parágrafo Primero

Del software y bases de datos

Apartado Primero Del software de código cerrado y bases de datos

Artículo 131.- Protección de software. - El software se protege como obra literaria. Dicha protección se otorga independientemente de que hayan sido incorporados en un ordenador y cualquiera sea la forma en que estén expresados, ya sea como código fuente; es decir, en forma legible por el ser humano; o como código objeto; es decir, en forma legible por máquina, ya sea sistemas operativos o sistemas aplicativos, incluyendo diagramas de flujo, planos, manuales de uso, y en general, aquellos elementos que conformen la estructura, secuencia y organización del programa. Se excluye de esta protección las formas estándar de desarrollo de software.

Artículo 132.- Adaptaciones necesarias para la utilización de software. - Sin perjuicio de los derechos morales del autor, el titular de los derechos sobre el software, o el propietario u otro usuario legítimo de un ejemplar del software, podrá realizar las adaptaciones necesarias para la utilización de este, de acuerdo con sus necesidades, siempre que ello no implique su utilización con fines comerciales**.**

Artículo 133.- Titulares de derechos. - Es titular de los derechos sobre un software el productor, esto es, la persona natural o jurídica que toma la iniciativa y responsabilidad de la realización de la obra. Se presumirá titular, salvo prueba en contrario, a la persona cuyo nombre conste en la obra o sus copias de la forma usual. Dicho titular está además autorizado para ejercer en nombre propio los derechos morales sobre la obra, incluyendo la facultad para decidir sobre su

divulgación. El productor tiene el derecho exclusivo de impedir que terceras personas realicen sin su consentimiento versiones sucesivas del software y software derivado del mismo. Las disposiciones del presente artículo podrán ser modificadas mediante acuerdo entre los autores y el productor.

Apartado Segundo De las tecnologías libres y formatos abiertos

Artículo 142.- Tecnologías libres. - Se entiende por tecnologías libres al software de código abierto, los estándares abiertos, los contenidos y el hardware libres.

Los tres primeros son considerados como Tecnologías Digitales Libres. Se entiende por software de código abierto al software en cuya licencia el titular garantiza al usuario el acceso al código fuente y lo faculta a usar dicho software con cualquier propósito.

Especialmente otorga a los usuarios, entre otras, las siguientes libertades esenciales:

- La libertad de ejecutar el software para cualquier propósito.
- La libertad de estudiar cómo funciona el software, y modificarlo para adaptarlo a cualquier necesidad.
- El acceso al código fuente es una condición imprescindible para ello;
- La libertad de redistribuir copias.
- La libertad de distribuir copias de sus versiones modificadas a terceros. Se entiende por código fuente, al conjunto de instrucciones escritas en algún lenguaje de programación, diseñadas con el fi n de ser leídas y transformadas por alguna herramienta de software en lenguaje de máquina o instrucciones ejecutables en la máquina. Los estándares abiertos son formas de manejo y almacenamiento de los datos en los que se conoce su estructura y se permite su modificación y acceso no imponiéndose ninguna restricción para su uso.

Artículo 146.- Localización de datos. - Cuando las entidades del sector público contraten servicios de software u otros que impliquen la localización de datos, deberán hacerlo con proveedores que garanticen que los datos se encuentren localizados en centros de cómputo que cumplan con los estándares internacionales en seguridad y protección conforme las siguientes reglas:

a) Los datos relacionados con seguridad nacional y sectores estratégicos se deben encontrar en centros de cómputo ubicados en territorio ecuatoriano;

b) Los datos de relevancia para el Estado que no se encuentren contenidos en el literal a) de este artículo se deben encontrar de forma preferente en centros

de cómputo ubicados en territorio ecuatoriano o en países con normas de protección de datos iguales o más exigentes que los establecidos en el Ecuador; y,

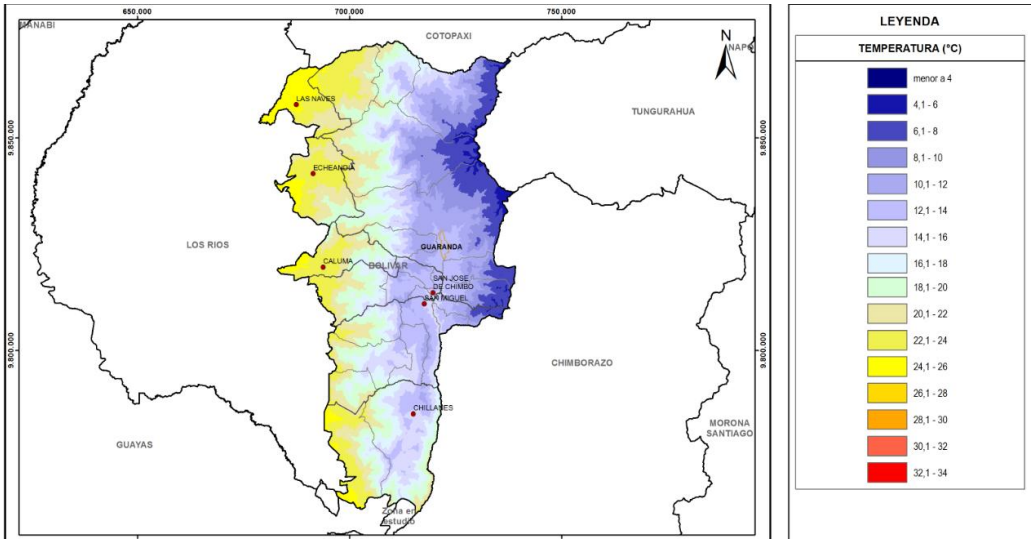
c) Los datos que no se encuentren contenidos en los literales a y b del presente artículo se deben encontrar de forma indistinta en centros de cómputo ubicados en territorio ecuatoriano o extranjero. (Asamblea Nacional del Ecuador, 2016)

Ley del Instituto de Meteorología e Hidrología" del INAMHI

Art. 15.- El Instituto Nacional de Meteorología e Hidrología y todas sus estaciones de observación gozarán de franquicia postal y telegráfica, y, en consecuencia, podrán utilizar para la transmisión de las observaciones, en el interior y el exterior, los servicios de radio, telefonía y telegrafía nacionales, y del servicio de correo aéreo y terrestre. (LEY DEL INSTITUTO DE METEOROLOGÍA E HIDROLOGÍA, INAMHI, n.d.)

2.5 Georreferencial

Figura 4. Mapa de temperatura de la Provincia Bolívar.



Fuente. (*16_TEMP_CP_2. Png (3306×2340)*, n.d.)

Ubicación de Bolívar en Ecuador Coordenadas: 1°37′00″S 79°07′00″O

3 CAPITULO III METODOLOGÍA

3.1 Metodología de Desarrollo de Software

La metodología utilizada para el desarrollo del proyecto se basa en el enfoque ágil SCRUM, que se caracteriza por su capacidad para manejar cambios y mejorar continuamente a lo largo del proceso de desarrollo. SCRUM utiliza un enfoque iterativo e incremental que permite la transparencia, inspección y adaptación, facilitando una mejor previsibilidad y gestión de riesgos.

Scrum, basado en el empirismo y el pensamiento Lean, utiliza un enfoque iterativo e incremental para mejorar la previsibilidad y gestionar riesgos. Los equipos en Scrum poseen las habilidades necesarias para realizar el trabajo, compartiendo y adquiriendo habilidades según sea necesario. Durante el Sprint, se realizan cuatro eventos formales para asegurar la transparencia, inspección y adaptación, pilares fundamentales del método. (Schwaber Ken & Sutherland Jeff, 2020)

A continuación, se describen las etapas del desarrollo del proyecto:

Etapa 1: Recolección y Procesamiento de Imágenes Satelitales

1. **Extracción Automática de Imágenes:**
    - **Desarrollo de Scripts de Descarga:**
        - Se crean scripts en Python que se ejecutan en la Raspberry Pi para conectarse al servidor de datos del satélite GOES-16.
        - Estos scripts automatizan la descarga de imágenes satelitales a intervalos regulares, garantizando la obtención continua de datos en tiempo real sin intervención manual.
    - **Programación de Tareas con Crontab:**
        - Se utiliza la herramienta crontab para programar la ejecución periódica de los scripts de descarga.
        - Por ejemplo, los scripts se configuran para ejecutarse cada 30 minutos utilizando una instrucción como */30 * * * *, sincronizando así la recolección de imágenes con las actualizaciones del satélite.

- **Almacenamiento Estructurado de Imágenes:**
  - Las imágenes descargadas se almacenan en una estructura de directorios organizada por fecha y hora, lo que facilita su identificación y acceso para el procesamiento posterior.
  - Se asegura que el espacio en disco se gestione eficientemente, implementando políticas de eliminación de imágenes antiguas si es necesario.

2. **Preprocesamiento de Imágenes:**
   - **Verificación de Integridad:**
     - Se implementan comprobaciones para asegurar que las imágenes descargadas no estén corruptas o incompletas.
     - Esto incluye verificar tamaños de archivo y formatos, asegurando que estén listas para el procesamiento.
   - **Normalización de Formatos:**
     - Si las imágenes vienen en diferentes formatos o resoluciones, se convierten a un formato estándar (como PNG) y se ajustan a una resolución uniforme.
     - Esto garantiza la consistencia en el procesamiento y la calidad de las animaciones generadas.

Etapa 2: Generación de Animaciones

1. **Creación de Aplicación para Generar Animaciones:**
   - **Procesamiento de Imágenes con OpenCV:**
     - **Recorte de la Región de Interés:**
       - Se utiliza OpenCV para recortar las imágenes y enfocar específicamente la región geográfica de Ecuador.
       - Esto elimina áreas no relevantes y reduce el tamaño de los archivos, optimizando el rendimiento del sistema.

- **Superposición de Capas Geográficas:**
  - Se añade una capa base que muestra las provincias y fronteras de Ecuador sobre las imágenes satelitales.
  - Esta superposición se realiza utilizando datos geoespaciales y permite una mejor interpretación visual de las imágenes por parte de los usuarios.
- **Ajustes de Imagen:**
  - Se aplican filtros y ajustes de contraste y brillo para mejorar la claridad y detalle de las imágenes.

- **Generación del Video:**
  - **Creación de Secuencia de Imágenes:**
    - Las imágenes procesadas se ordenan cronológicamente para crear una secuencia lógica que refleje cambios en el tiempo.
  - **Conversión a Video:**
    - Se utiliza OpenCV o herramientas como FFmpeg para convertir la secuencia de imágenes en un archivo de video.
    - Se configuran parámetros como la tasa de fotogramas (frames por segundo) para asegurar una animación fluida.
  - **Optimización del Tamaño del Video:**
    - Se ajustan la resolución y la compresión del video para equilibrar la calidad visual y el tamaño del archivo.
    - Esto asegura que el video sea fácilmente transferible y reproducible en la web sin tiempos de carga excesivos.

2. **Creación de un Cliente FTP:**
    - **Desarrollo del Script de Carga Automática:**
        - Se implementa un script en Python que funciona como cliente FTP para conectarse al servidor web de grintec.org.
        - El script automatiza la transferencia del archivo de video generado al servidor, eliminando la necesidad de carga manual.
    - **Programación con Crontab:**
        - El script de carga se programa para ejecutarse automáticamente después de que se genera el video.
        - Esto se logra mediante crontab, coordinando la secuencia de generación y carga para mantener las animaciones actualizadas en el servidor.
    - **Gestión de Errores y Reintentos:**
        - Se implementan mecanismos para detectar errores en la transferencia, como fallos de conexión o interrupciones.
        - En caso de error, el script registra el incidente y puede reintentar la transferencia después de un período determinado.

Etapa 3: Desarrollo del Módulo Web

1. **Visualización de Animaciones:**
    - **Desarrollo del Visor de Animaciones:**
        - Se crea una aplicación web utilizando Node.js y el framework Express.js para servir la interfaz del visor de animaciones en grintec.org.
        - El visor permite a los usuarios reproducir las animaciones directamente desde el navegador sin necesidad de descargar los archivos.

- **Integración del Reproductor de Video:**
  - Se incorpora un reproductor de video HTML5 en la página web que soporta formatos compatibles y ofrece controles de reproducción.
  - Se aseguran funcionalidades como reproducir, pausar, ajustar el volumen y cambiar a pantalla completa.
- **Navegación entre Fechas y Horas:**
  - Se implementa un sistema de selección que permite a los usuarios elegir entre diferentes fechas y horas para visualizar las animaciones correspondientes.
  - Esto puede incluir menús desplegables, calendarios interactivos o barras de desplazamiento temporal.
- **Diseño de Interfaz Intuitiva:**
  - Se presta especial atención a la usabilidad y experiencia de usuario, diseñando una interfaz limpia y fácil de navegar.
  - Se utilizan principios de diseño responsivo para asegurar que el visor funcione correctamente en dispositivos de escritorio y móviles.

2. **Pruebas:**
   - **Pruebas Funcionales:**
     - Se verifica que todas las funcionalidades de la aplicación web y los scripts de procesamiento funcionen según lo esperado.
     - Esto incluye comprobar que las animaciones se generan correctamente, que se cargan al servidor y que son accesibles desde la web.
   - **Pruebas de Usabilidad:**
     - Se realizan pruebas con usuarios para evaluar la facilidad de uso del visor y la claridad de la información presentada.

        - Se recopila retroalimentación y se realizan ajustes en la interfaz y las funcionalidades para mejorar la experiencia del usuario.

    - **Pruebas de Rendimiento:**

        - Se evalúa el tiempo de carga de las animaciones y la capacidad de la aplicación para manejar múltiples usuarios simultáneos.

        - Se optimizan aspectos como la compresión de archivos y el uso de caché para mejorar el rendimiento.

    - **Documentación y Mantenimiento:**

        - Se documenta detalladamente el sistema, incluyendo los scripts, configuraciones y procedimientos, facilitando el mantenimiento y futuras actualizaciones.

        - Se establecen protocolos para el monitoreo continuo del sistema y la resolución de posibles problemas que puedan surgir.

3.2 Técnicas e Instrumentos de Recopilación de Datos

La selección y elaboración de técnicas e instrumentos son cruciales para la recolección de información durante el proceso investigativo de nuestro proyecto de generación de timelapses a partir de imágenes satelitales.

Se ha seleccionado el formato de entrevista como el método principal de comunicación. Las entrevistas facilitan el contacto directo con el director y los colaboradores de la investigación, permitiendo abordar y analizar los requisitos en tiempo real durante cada reunión. Esta técnica asegura una comprensión clara y precisa de las necesidades y expectativas del proyecto. (Folgueiras Bertomeu, 2016)

Se utilizarán fichas de observación para llevar un registro ordenado de los datos más importantes durante el procesamiento y generación de timelapses. Este método permitirá evaluar el avance de los resultados obtenidos y ajustar los procesos según sea necesario. La observación directa del proceso técnico ayudará a identificar posibles áreas de mejora y asegurar que los objetivos del proyecto se cumplan eficientemente.

4 CAPITULO IV INGENIERÍA DEL PROYECTO

4.1 Análisis

4.1.1 Especificación de Requerimientos de Software

4.1.1.1 Introducción

Este documento es una Especificación de Requisitos Software (ERS) para el sistema de generación y visualización de animaciones a partir de imágenes satelitales del GOES-16 en la provincia de Bolívar. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

4.1.1.2 Alcance del Producto

El sistema para generar Time Lapse a partir de imágenes extraídas del GOES-16 proporciona una plataforma de información en línea para la generación y visualización de animaciones de datos climáticos en tiempo real de la provincia de Bolívar, incluyendo:

- **Generación de animaciones**: Procesamiento de imágenes satelitales y superposición de capas para crear videos time lapse.
- **Subida automática:** Carga de los videos generados a un servidor web.
- **Visualización en línea:** Interfaz web que permite la visualización interactiva y ordenada de las animaciones generadas.

4.1.1.3 Definiciones, Acrónicos y Abreviaturas

Tabla *2* Definiciones de acrónimos, siglas y abreviaturas usadas en el documento ERS

NOMBRE	DESCRIPCIÓN
GOES-16	Geostationary Operational Environmental Satellite 16, un satélite meteorológico operado por NOAA.
Satélites Meteorológicos	Satélites que llevan instrumentos para observar la Tierra y su atmósfera.
Sensores Remotos	Dispositivos electrónicos diseñados para detectar y medir características del entorno desde una distancia, como temperatura, humedad, presión atmosférica, radiación solar, entre otros, mediante la captación de señales electromagnéticas.

Imágenes Satelitales	Fotografías o datos visuales capturados por satélites artificiales en órbita alrededor de la Tierra
Rpi	Raspberry Pi, una serie de computadoras de placa única de bajo costo.
OpenCV	Open Source Computer Vision Library, una biblioteca de visión artificial y procesamiento de imágenes.
HTTP	Protocolo de Transferencia de Hipertexto, base de la comunicación de datos en la web.
JSON	Notación de Objetos de JavaScript, un formato de intercambio de datos ligero.
HTML	Lenguaje de Marcado de Hipertexto, el estándar para la creación de páginas web.
CSS	Hojas de Estilo en Cascada, utilizadas para diseñar y maquetar páginas web.
JavaScript	Lenguaje de programación utilizado para crear contenido dinámico en las páginas web.
IDE	Entorno de Desarrollo Integrado, una aplicación que proporciona herramientas y servicios integrales para el desarrollo de software.
QA	Aseguramiento de la calidad, actividades que garantizan la calidad del producto final.
ZULU	Término utilizado en aviación, navegación y operaciones militares para referirse al Tiempo Universal Coordinado (UTC)
SDR	Radio Definida por Software utiliza software para realizar funciones de procesamiento de señales de radio, ofreciendo flexibilidad y actualización fácil en comparación con hardware tradicional.
FTP	Protocolo de Transferencia de Archivos, utilizado para la transferencia de archivos entre sistemas.
Expressjs	Un framework web para Node.js, diseñado para construir aplicaciones web y APIs de manera eficiente.
Nodejs	Entorno de ejecución de JavaScript que permite ejecutar código JavaScript del lado del servidor.
PM2	Administrador de procesos para Node.js que facilita la gestión de aplicaciones en producción.

Crontab	Herramienta de Unix/Linux que permite programar la ejecución de scripts o comandos a intervalos específicos.
Superresolución	Técnica de procesamiento de imágenes utilizada para aumentar la resolución de una imagen más allá de su calidad original.
MoviePy	Biblioteca Python para edición de videos que permite crear, modificar y exportar clips de video.
FSRCNN	Modelo de redes neuronales convolucionales para superresolución rápida, utilizado para mejorar la calidad de las imágenes.
Canal/Banda ABI	Se refiere a las distintas frecuencias del espectro electromagnético que el satélite **GOES-16** utiliza a través del instrumento **ABI (Advanced Baseline Imager)** para capturar diferentes tipos de imágenes climáticas y atmosféricas. Los usuarios del sistema pueden seleccionar distintos canales o bandas ABI para ver las imágenes correspondientes.

4.1.1.4 Referencias

Tabla 3 Referencias

Título Del Documento	Referencia
Standard IEEE 830-1998	IEEE
IEC 20246-2017	ISO

Elaborado por: Bryan Gaibor y Franklin García

4.1.2 Descripción General del Producto

4.1.2.1 Perspectiva del producto

El sistema se compone de dos scripts principales que se ejecutan en una Raspberry Pi 5 y una plataforma web desarrollada con nodejs que se despliega en el servidor www.grintec.org. Los scripts automatizan el procesamiento de imágenes satelitales del GOES-16 y la generación de videos time lapse. Además, gestionan la subida automática de estos videos a un servidor FTP. La plataforma web, basada en Node.js y Express.js, permite a los usuarios visualizar las animaciones de manera interactiva y cronológica a través de un visor web optimizado.

4.1.2.2 Funciones del producto

Generación de Animaciones:

- Procesamiento de imágenes descargadas del satélite **GOES-16** para generar videos en formato **time lapse**.
- Superposición de capas geográficas que muestran las provincias del Ecuador en las imágenes.
- Compresión y optimización del video generado para mejorar la eficiencia en el almacenamiento y transmisión.

Subida Automática de Videos:

- Los videos generados se suben automáticamente a un servidor FTP, donde se almacenan para su visualización posterior.
- Los videos están organizados en carpetas por canal o banda del satélite GOES-16.

Visualización en Línea:

- Interfaz web interactiva que permite a los usuarios visualizar los videos time lapse.
- Los usuarios pueden seleccionar diferentes canales o bandas del satélite, lo que cambiará el conjunto de videos disponibles para su visualización.
- Navegación cronológica de los videos, puede desplazar entre fechas de los videos del canal que el usuario haya seleccionado.
- Funcionalidad para reproducir las animaciones generadas a partir de imágenes satelitales, con controles para pausar, adelantar o retroceder las animaciones.

4.1.2.3 Características de los usuarios

Tabla 4 Tipo de Usuario - Usuario

Tipo de usuario	Usuario
Características	Personas interesadas en información climática y visualización de animaciones satelitales, con conocimientos básicos en navegación en internet
Actividades	Acceder y visualizar animaciones del clima generadas a partir de imágenes satelitales del GOES-16, interpretar datos climáticos, y enviar sugerencias o consultas a través de la plataforma web.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 5 Tipo de Usuario - Administrador

Tipo de usuario	Administrador
Características	Personal técnico del proyecto GRINTEC encargado de la gestión y mantenimiento del sistema.
Actividades	Monitorear y asegurar el correcto funcionamiento de los scripts que generan y suben las animaciones, gestionar la infraestructura de hardware y software, y realizar tareas de mantenimiento y actualización del sistema.

Elaborado por: Bryan Gaibor y Franklin García

4.1.2.4 Restricciones generales

- **Conectividad:** Requiere conexión a internet para la descarga de imágenes y la subida de videos.
- **Compatibilidad:** La interfaz web debe ser compatible con los navegadores más utilizados, debe soportar HTML5 o superior.

4.1.3 Requerimientos de Usuario

4.1.3.1 Requerimientos funcionales

Tabla 6 Requerimiento funcional 01: Selección de Canal y Fecha

Identificación del Requerimiento	**RF01**
Prioridad	Alta
Nombre del Requerimiento	Selección de Canal y Fecha
Actores	Usuario
Descripción del Requerimiento	Los usuarios podrán seleccionar el canal (banda espectral del satélite GOES-16) y la fecha desde la interfaz web para visualizar las animaciones.
Precondiciones	Tener acceso a Internet y un dispositivo compatible con navegadores web modernos.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 7 Requerimiento funcional 02: Visualización de animaciones

Identificación del Requerimiento	**RF02**
Prioridad	Alta
Nombre del Requerimiento	Visualización de Animaciones
Actores	Usuario
Descripción del Requerimiento	El sistema permitirá visualizar animaciones generadas a partir de imágenes satelitales, con superposición de capas geográficas que muestran las provincias de Ecuador.
Controles	El usuario podrá pausar, adelantar, retroceder y cambiar a pantalla completa.

Elaborado por: Bryan Gaibor y Franklin García

4.1.3.2 Requerimientos no funcionales

Tabla 8 Requerimientos no funcionales 01: Funcionalidad

Identificación del Requerimiento	**RNF01**
Prioridad	Alta
Nombre del Requerimiento	Funcionalidad
Descripción del Requerimiento	El sistema debe operar correctamente en todo momento, garantizando que las animaciones se generen, procesen y suban sin errores.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 9 Requerimientos no funcionales 02: Usabilidad

Identificación del Requerimiento	**RNF02**
Prioridad	Alta
Nombre del Requerimiento	Usabilidad
Descripción del Requerimiento	La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo a los usuarios acceder a las animaciones sin necesidad de conocimientos técnicos avanzados.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 10 Requerimientos no funcionales 03: Disponibilidad

Identificación del Requerimiento	**RNF03**
Prioridad	Alta
Nombre del Requerimiento	Disponibilidad
Descripción del Requerimiento	La plataforma debe estar disponible para los usuarios en todo momento, con tiempos de inactividad mínimos.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 11 Requerimientos no funcionales 04: Compatibilidad

Identificación del Requerimiento	**RNF04**
Prioridad	Media
Nombre del Requerimiento	Compatibilidad
Descripción del Requerimiento	La plataforma debe ser compatible con los navegadores web más utilizados, como Google Chrome, Firefox y Microsoft Edge, y debe soportar HTML5.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 12 Requerimientos no funcionales 05: Rendimiento

Identificación del Requerimiento	**RNF05**
Prioridad	Media
Nombre del Requerimiento	Rendimiento
Descripción del Requerimiento	El sistema debe generar y cargar las animaciones en menos de 5 minutos después de recibir las imágenes del satélite.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 13 Requerimientos no funcionales 06: Seguridad

Identificación del Requerimiento	**RNF06**
Prioridad	Alta
Nombre del Requerimiento	Seguridad
Descripción del Requerimiento	Los datos y videos transferidos deben estar protegidos mediante medidas de seguridad, como autenticación en el FTP y cifrado de datos sensibles.

Tabla 14 Requerimientos no funcionales 07: Generación Automática de Animaciones

Identificación del Requerimiento	**RNF07**
Prioridad	Alta
Nombre del Requerimiento	Generación Automática de Animaciones
Actores	Sistema
Descripción del Requerimiento	El sistema debe procesar automáticamente las imágenes satelitales del GOES-16 para crear animaciones, incluyendo recorte y superposición de capas geográficas.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 15 Requerimientos no funcionales 08: Carga Automática al Servidor.

Identificación del Requerimiento	**RNF08**
Prioridad	Alta
Nombre del Requerimiento	Carga Automática al Servidor
Actores	Sistema
Descripción del Requerimiento	Los videos generados deben ser cargados automáticamente al servidor web mediante un cliente FTP.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 16 Requerimientos no funcionales 09: Automatización del recorte y envió de archivos

Identificación del Requerimiento	**RNF09**
Prioridad	Alta
Nombre del Requerimiento	Automatización del Recorte y Envío de Archivos
Actores	Sistema
Descripción del Requerimiento	El sistema debe automatizar el recorte de imágenes satelitales y el envío de los archivos generados al servidor mediante funciones cron en Linux. Estas tareas deben ejecutarse de manera periódica y sin intervención manual.
Precondiciones	El cron debe estar configurado correctamente en el sistema operativo.

Elaborado por: Bryan Gaibor y Franklin García

4.1.3.3 Requerimientos de interfaz

Tabla 17 Requerimiento de interfaz 01: Interfaz de usuario

Identificación del Requerimiento	**RI01**
Prioridad	Alta
Nombre del Requerimiento	Interfaz de Usuario
Descripción del Requerimiento	La interfaz web debe ser intuitiva y fácil de navegar, con menús y botones claramente etiquetados, y un diseño responsivo que se adapte a diferentes tamaños de pantalla. La interfaz gráfica debe ser agradable a la vista y permitir al usuario identificar rápidamente los componentes del sistema.

Tabla 18 Requerimiento de interfaz 02: Compatibilidad de navegadores

Identificación del Requerimiento	**RI02**
Prioridad	Media
Nombre del Requerimiento	Compatibilidad de Navegadores
Descripción del Requerimiento	La plataforma debe ser compatible con los navegadores web más comunes, incluyendo Google Chrome, Firefox, y Microsoft Edge, para asegurar el acceso universal de los usuarios.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 19 Requerimiento de interfaz 03: Interfaz de hardware

Identificación del Requerimiento	**RI03**
Prioridad	Alta
Nombre del Requerimiento	Interfaz de Hardware
Descripción del Requerimiento	El sistema debe funcionar en una Raspberry Pi para el procesamiento y generación de las animaciones. La Raspberry Pi debe tener una tarjeta de red, procesador de al menos 1GHz, memoria RAM de al menos 1GB, y almacenamiento de al menos 8GB.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 20 Requerimiento de interfaz 04: Interfaz de software

Identificación del Requerimiento	**RI04**
Prioridad	Alta
Nombre del Requerimiento	Interfaz de Software

Descripción del Requerimiento	La plataforma debe ser compatible con los sistemas operativos y software necesarios para la Raspberry Pi, incluyendo Raspbian y Python. Además, debe permitir el envío de archivos vía FTP al servidor web donde está alojada la web grintec.org.

Elaborado por: Bryan Gaibor y Franklin García

4.2 Diseño

4.2.1 Arquitectura del Software

La arquitectura de la aplicación web se basa en el patrón arquitectónico cliente-servidor, lo que permite una estructura modular, organizada y de fácil gestión. Este enfoque de diseño fomenta la separación de responsabilidades, facilitando actualizaciones y mejoras en componentes específicos de la aplicación sin afectar otras partes.

La arquitectura cliente-servidor implementada en el proyecto distingue claramente la lógica de negocio en el servidor, la presentación en el cliente y la comunicación entre ambos. Esto optimiza el flujo de información, garantiza la eficiencia en la gestión de datos y simplifica el mantenimiento del sistema.

A continuación, se detallan las operaciones de cada nivel de la arquitectura:

1. **Nivel Cliente (Raspberry Pi):** Este nivel es responsable del procesamiento inicial de los datos en el dispositivo cliente, el cual ejecuta los siguientes scripts en Python para la preparación y transmisión de archivos multimedia:

    - **recorte.py**: Procesa localmente las imágenes, aplicando recortes específicos según los parámetros definidos.
    - **enviar_ftp.py**: Gestiona la transferencia de los archivos procesados al servidor remoto utilizando el protocolo FTP.

    El cliente actúa como un punto de origen para la creación y el envío de los datos al servidor, asegurando que los recursos multimedia estén listos para su almacenamiento y publicación.

2. **Nivel Servidor:** En este nivel, el servidor web desempeña un papel central al recibir, gestionar y almacenar los archivos enviados por el cliente. Las operaciones específicas incluyen:

    - **Recepción y almacenamiento de videos**: Los videos transferidos mediante FTP son gestionados y organizados adecuadamente en el servidor para su disponibilidad en el sitio web.
    - **Servidor web con Express**: Gestiona las solicitudes HTTP y HTTPS, permitiendo a los usuarios acceder y visualizar los videos desde el dominio www.grintec.org.

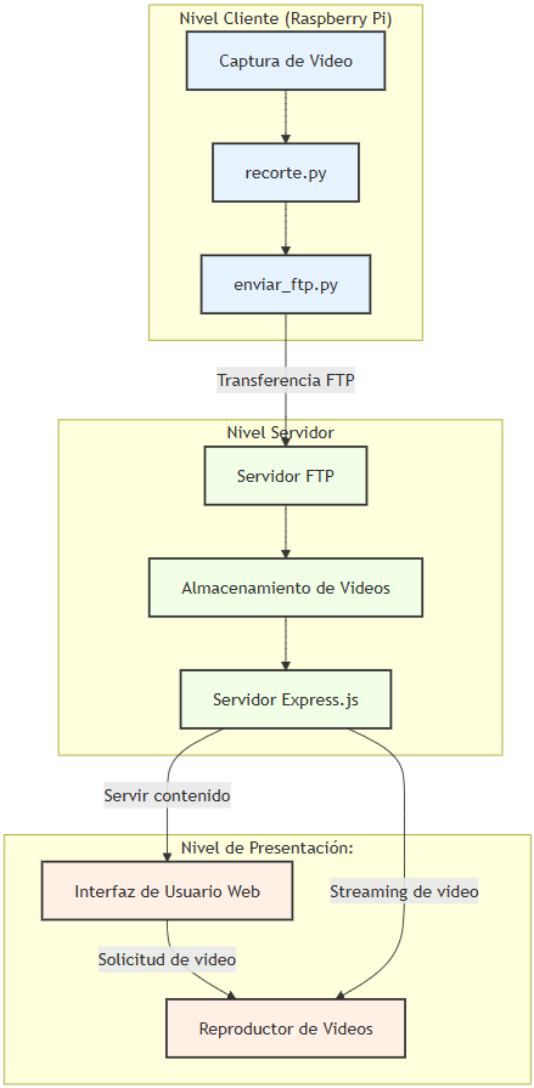
    Esta capa intermedia asegura que los recursos multimedia se mantengan organizados y accesibles, proporcionando un entorno seguro para la distribución de datos.

3. **Nivel de Presentación (Sitio web www.grintec.org):** Este nivel se encarga de proporcionar una interfaz amigable y accesible a los usuarios finales, desde donde pueden interactuar y visualizar los videos procesados:

    - **Interfaz de usuario**: El sitio web permite a los usuarios explorar y reproducir los videos almacenados en el servidor, facilitando una experiencia de navegación intuitiva y ágil.
    - **Entrega de contenido**: El servidor Express sirve los archivos de video al navegador del usuario, gestionando eficientemente las solicitudes y asegurando una reproducción fluida y responsiva.

    Este nivel funciona como el punto de interacción entre los usuarios y los recursos multimedia, ofreciendo una plataforma centralizada y accesible.

Figura 5. Arquitectura cliente-servidor

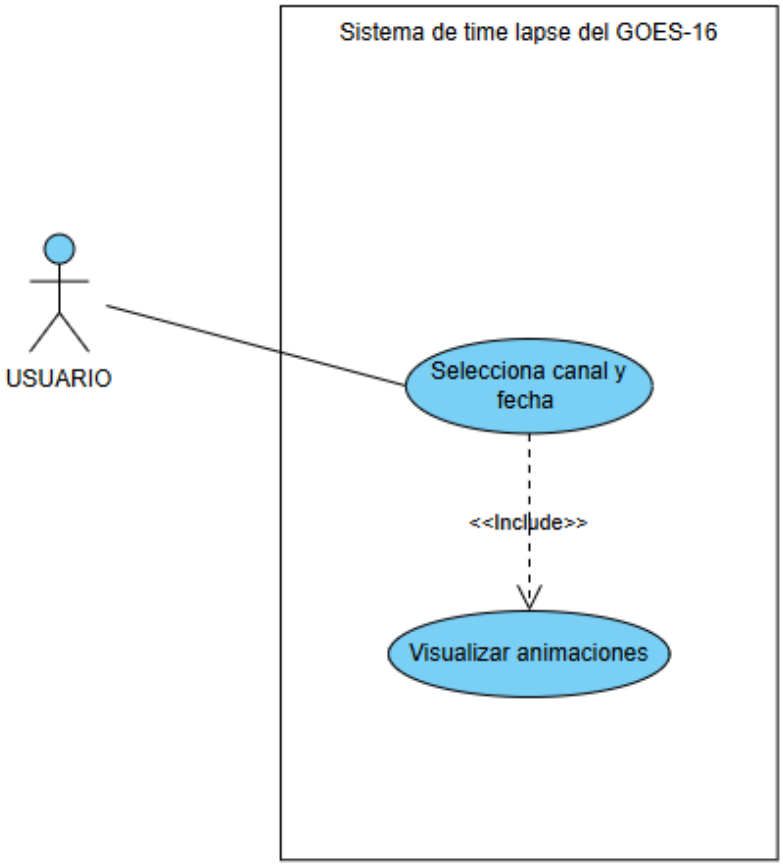


Elaborado por: Bryan Gaibor y Franklin García

4.2.2 Diagramas

4.2.2.1 Diagrama de Caso de Uso

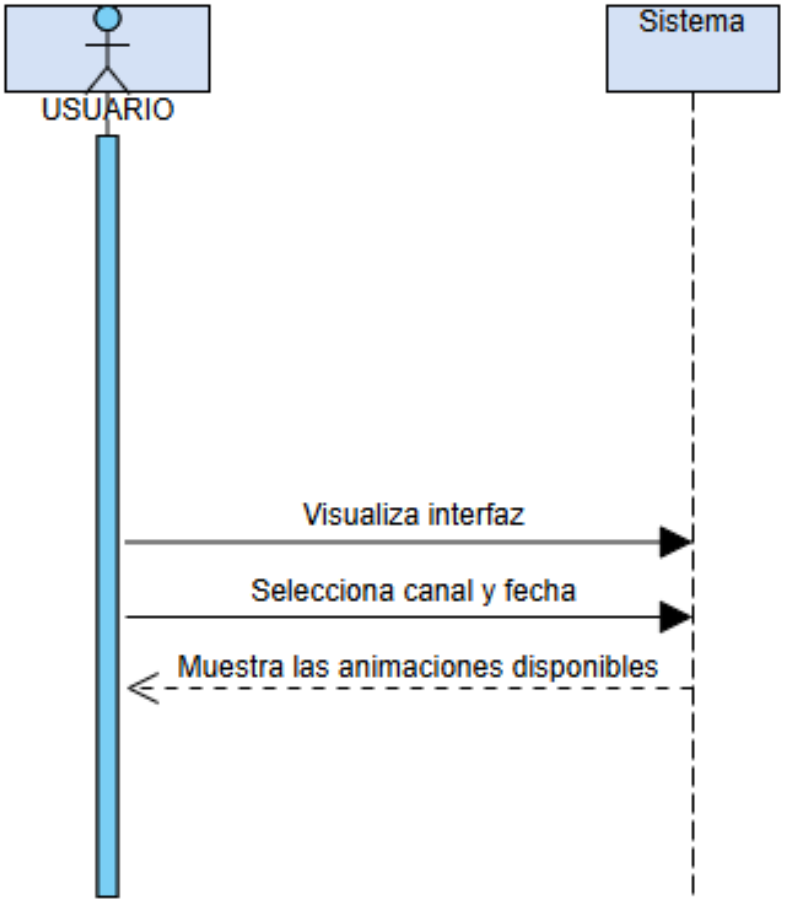
Figura 6. Diagrama de Casos de Uso



Elaborado por: Bryan Gaibor y Franklin García

4.2.2.2 Diagrama de Secuencia

Figura 7 Diagrama de secuencia

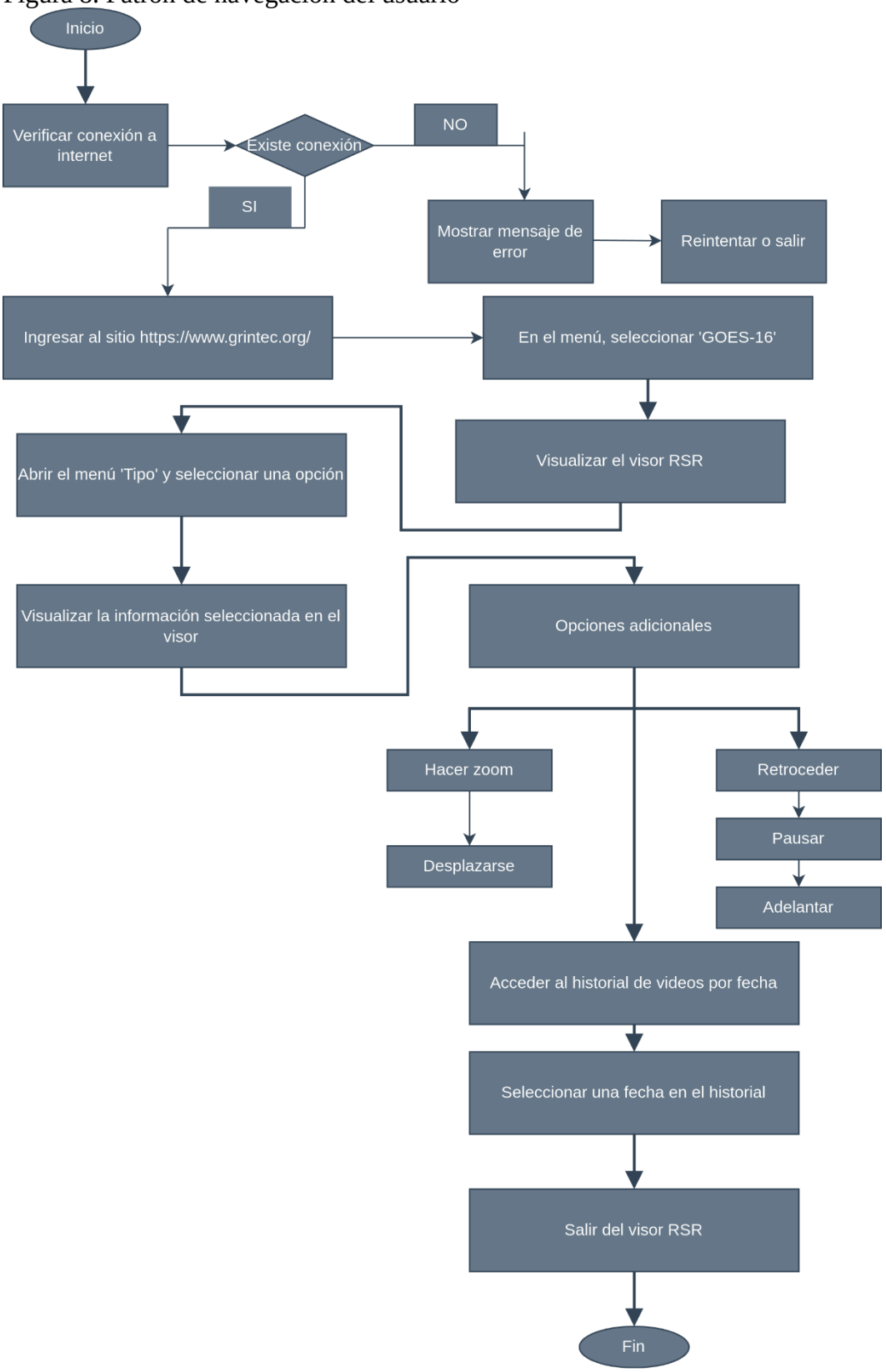


Elaborado por: Bryan Gaibor y Franklin García

4.2.3 Diseño de la Interfaz

4.2.3.1 Patrón de navegación

Figura 8. Patrón de navegación del usuario



Elaborado por: Bryan Gaibor y Franklin García

4.2.3.2 Interfaz de Pantallas

Color

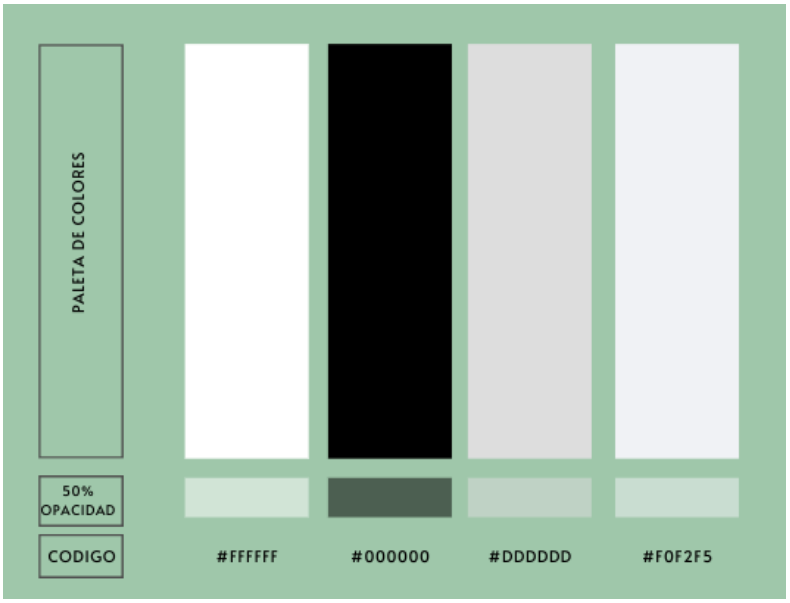
La elección del color blanco para nuestra página web ha resultado ser una decisión exitosa por varias razones. En primer lugar, el blanco evoca sensaciones de simplicidad y transparencia, lo que fortalece la claridad y la accesibilidad de nuestro sitio. Además, este color es conocido por transmitir frescura y amplitud, lo que crea una experiencia de usuario más ordenada y agradable. También, el blanco mejora la legibilidad y el contraste, garantizando que el contenido y la información sean fácilmente perceptibles.

Optar por el blanco ha dado como resultado un diseño limpio, atractivo y funcional para nuestra página web, mejorando tanto la satisfacción como la experiencia del usuario.

En nuestra pantalla se presentan los siguientes colores:

- **Color de fondo:** #ffffff (Blanco)
- **Color primario:** #000000 (Negro)
- **Color secundario:** #dddddd (Gris claro)
- **Color de fondo del cuerpo:** #f0f2f5 (Gris muy claro)
- **Color de fondo de la barra lateral:** #ffffff (Blanco)
- **Color del borde derecho de la barra lateral:** #dddddd (Gris claro)

Figura 9. Paleta de colores

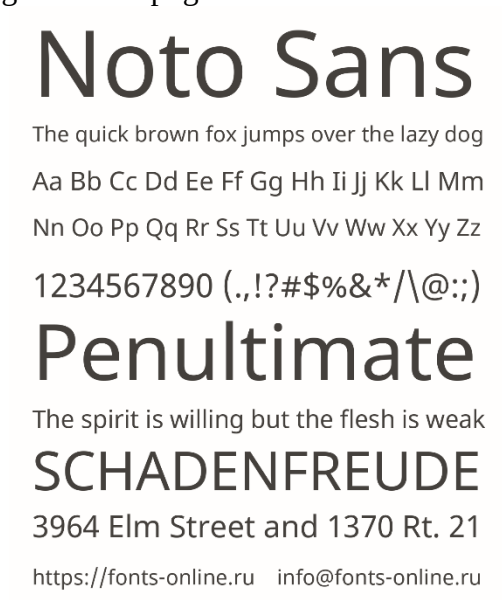


Elaborador por: Bryan Gaibor y Franklin García.

Tipografía

Noto Sans es una tipografía sans-serif contemporánea y flexible, creada por Google como parte del proyecto Noto. Este proyecto busca ofrecer un conjunto de fuentes que abarque todos los idiomas y sistemas de escritura del mundo. Destaca por su claridad y diseño minimalista, lo que la hace ideal para una amplia gama de usos.

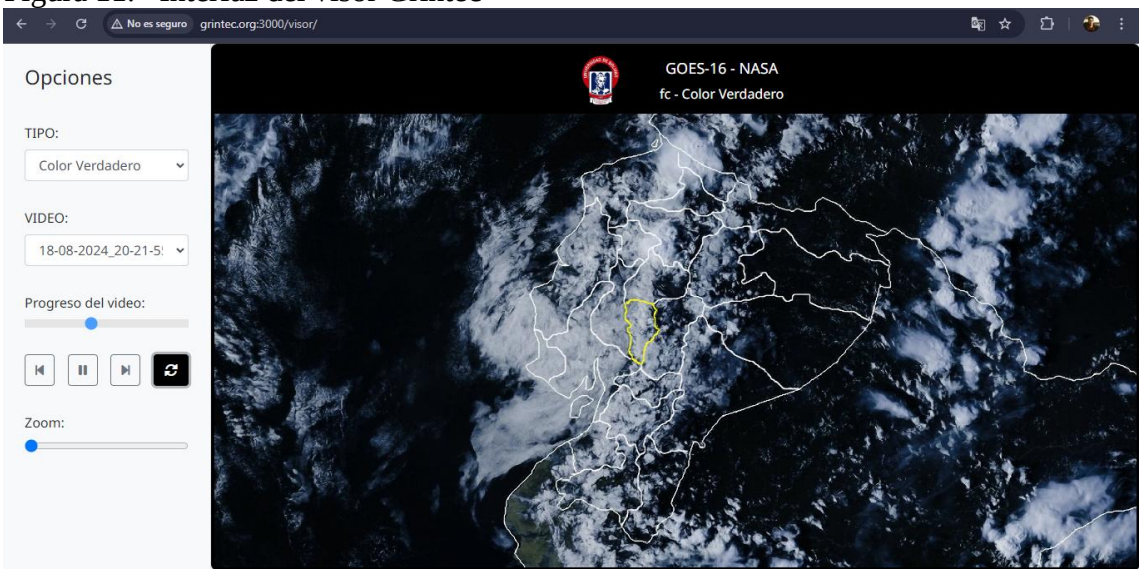
Figura 10. Tipografía Noto Sans



Fuente: Online Fonts. (n.d.). Noto Sans. Online Fonts.

4.2.3.3 Interfaz de visor Grintec

Figura 11. Interfaz del visor Grintec



Elaborador por: Bryan Gaibor y Franklin García.

4.3 Programación

4.3.1 Definición del objetivo

Desarrollar una aplicación que genere animaciones de tipo Time Lapse centradas en la provincia de Bolívar, utilizando datos e imágenes en tiempo real del satélite GOES-16. Esta herramienta informática se enfocará en procesar las imágenes de manera eficiente, garantizando que no se pierda información relevante durante la generación de las animaciones. Además, se creará una aplicación para subir las imágenes al servidor grintec.org, facilitando la gestión y el acceso a los datos satelitales procesados. Finalmente, se integrará un módulo web que permita la visualización de las animaciones generadas, ofreciendo a los usuarios un acceso sencillo y directo a la información.

4.3.2 Análisis del problema

Desarrollar una aplicación que genera animaciones de tipo Time Lapse centradas en la provincia de Bolívar podría abordar una serie de desafíos en el procesamiento y visualización de datos satelitales:

- **Procesamiento en tiempo real de imágenes**: Esta aplicación podría optimizar el procesamiento de imágenes satelitales en tiempo real, asegurando que se conserve toda la información relevante durante la creación de animaciones. Esto permitiría la obtención de representaciones visuales precisas y actualizadas de fenómenos climáticos y otros eventos en la provincia.
- **Centralización y acceso a datos**: Al integrar la capacidad de subir imágenes al servidor grintec.org, la aplicación ofrecería una plataforma única para gestionar y acceder a los datos satelitales procesados, simplificando la consulta y el análisis de esta información.
- **Visualización intuitiva**: Con el módulo web integrado, los usuarios podrán visualizar las animaciones generadas de manera accesible y directa. Esto facilitaría la comprensión de patrones y cambios a lo largo del tiempo, ayudando a tomar decisiones basadas en datos visuales precisos.

- **Planificación y gestión de recursos**: Las animaciones podrían servir para planificar actividades al aire libre, gestionar recursos agrícolas y tomar decisiones estratégicas fundamentadas en las visualizaciones proporcionadas por la aplicación.
- **Preparación para eventos climáticos**: Las animaciones serían útiles para anticipar y responder a eventos climáticos severos, ofreciendo una visualización clara y en tiempo real de las alteraciones en el entorno.
- **Conciencia ambiental**: La aplicación promovería la conciencia ambiental al presentar visualizaciones detalladas de patrones climáticos y cambios temporales, contribuyendo así a la conservación y a la toma de decisiones más sostenibles.

4.3.3 Diseño del Algoritmo

1. Verificar la conexión a internet
2. Ingresar al sitio https://www.grintec.org/
3. En el menú, seleccionar "GOES-EC"
4. Visualizar el visor Grintec.
5. Abrir el menú "Tipo " y seleccionar una de las siguientes opciones:
   - Visible rojo
   - Infrarrojo de onda corta
   - Infrarrojo de onda corta mejorado
   - Vapor de agua nivel alto
   - Vapor de agua nivel alto mejorado
   - Vapor de agua nivel medio
   - Vapor de agua nivel medio mejorado
   - Infrarrojo (banda limpia)
   - Infrarrojo (banda limpia mejorado)
   - Infrarrojo onda larga
   - Infrarrojo onda larga mejorado
   - Infrarrojo (banda sucia)
   - Infrarrojo banda sucia mejorada
   - Color verdadero
6. Visualizar la información seleccionada en el visor
7. Opciones adicionales:

- Pausar
- Retroceder
- Adelantar
- Ampliar y Reducir la imagen
- Desplazarse
- Loop

8. Acceder al historial de videos
9. Seleccionar una fecha en el historial
10. Salir del visor Grintec

4.4 Codificación

Nuestro proyecto emplea una serie de herramientas que trabajan en conjunto para procesar, mejorar y publicar datos satelitales en tiempo real, generando animaciones climáticas para la provincia de Bolívar basadas en información del satélite GOES-16. Este proceso se logra mediante la ejecución coordinada de dos scripts clave, recorte.py y enviar_ftp.py, junto con tecnologías web.

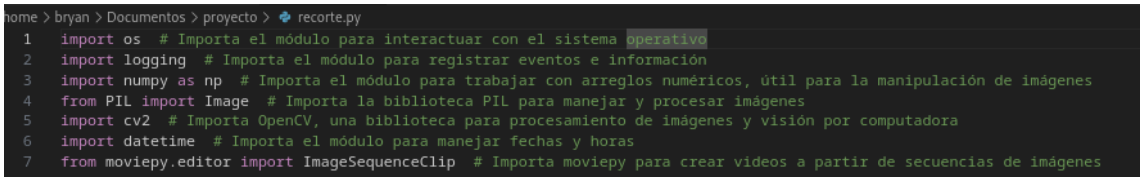
Procesamiento de imágenes (recorte.py)

Este script es responsable de recibir y manipular imágenes satelitales, realizando varias tareas esenciales:

Recorte y superposición: Extrae secciones específicas de cada imagen, enfocándose en áreas de interés, y añade una capa de límites provinciales para facilitar su comprensión geográfica.

Mejora de calidad: Aplica un modelo avanzado de superresolución (FSRCNN), mejorando la calidad de las imágenes con un aumento de resolución de 4x, de modo que se logre la claridad visual necesaria para el análisis.
Creación de video: Las imágenes procesadas se integran en un video (time-lapse) que permite visualizar de forma continua los cambios climáticos.
En los scripts, hemos incorporado varias librerías que son esenciales para el procesamiento de imágenes y la creación de videos.

Figura 12. Librerías utilizadas en script recorte.py



Elaborado por: Bryan Gaibor y Franklin García.

- **OS** nos brinda la capacidad de manejar rutas y archivos en el sistema, lo que hace más fácil la gestión de directorios y la eliminación de archivos temporales.
- **cv2 (OpenCV)** es una herramienta poderosa en el ámbito de la visión por computadora, ofreciendo funcionalidades avanzadas que nos

permiten realizar tareas como el recorte de imágenes y la mejora de su calidad a través de técnicas de superresolución.

- **MoviePy** se emplea para generar videos a partir de secuencias de imágenes, facilitando así la creación de contenido audiovisual a partir de nuestros datos.
- **logging** proporciona un mecanismo de registro que ayuda a capturar y manejar eventos del sistema, siendo crucial para la depuración y el seguimiento de errores durante la ejecución de los scripts.
- **configparser** simplifica la carga de configuraciones desde archivos externos, lo que hace que nuestro código sea más adaptable y flexible para diferentes entornos.
- **glob** facilita la búsqueda de archivos en un directorio según patrones específicos, lo que resulta útil para seleccionar imágenes en el proceso de creación de videos.
- Numpy es fundamental para trabajar con arreglos numéricos, lo que resulta útil en la manipulación de imágenes y la realización de cálculos matemáticos necesarios en el procesamiento de datos.
- PIL (Python Imaging Library) nos permite manejar y procesar imágenes de manera eficiente, ofreciendo herramientas para abrir, manipular y guardar imágenes en varios formatos.

Estas librerías colaboran de manera efectiva para llevar a cabo nuestras tareas, desde el manejo de imágenes hasta la producción de contenido multimedia.

Una vez que hemos revisado todos los aspectos importantes de nuestro proyecto, es hora de enfocarnos en las funciones clave del código. En esta parte, nos dedicaremos a examinar cómo se activan estas funciones, es decir, cómo están estructuradas y organizadas para alcanzar los objetivos del script. Este análisis nos ayudará a entender mejor la lógica detrás de la implementación y a apreciar la conexión entre las distintas secciones del código.

- **read_config():** Esta función se encarga de cargar la configuración desde un archivo externo llamado config.txt. Abre el archivo y lee cada línea, omitiendo las que están vacías o son comentarios. Los pares de

clave-valor que obtiene se almacenan en un diccionario, el cual se utilizará a lo largo del programa.

- **process_images(image_dir, cropped_dir):** Esta función procesa las imágenes de un directorio específico. Realiza recortes basados en coordenadas predefinidas y añade una capa adicional de información. Cada imagen se convierte al formato OpenCV y se mejora su calidad utilizando el modelo de superresolución FSRCNN antes de ser guardadas en un directorio temporal.
- **create_video(video_name_mp4, cropped_dir, output_dir):** Se encarga de recopilar las imágenes procesadas en el orden correcto para crear un video en formato MP4. Utiliza MoviePy para generarlo a una velocidad de un cuadro por segundo. El archivo final se guarda en el directorio de salida que se haya especificado.
- **clean_up(cropped_dir):** Esta función elimina todos los archivos en el directorio temporal y también borra la carpeta una vez que se ha creado el video. Si surge algún error durante este proceso, se registra un mensaje de error para su seguimiento.
- **main():** Aquí se gestiona el flujo principal del programa. Se verifica la existencia de las carpetas de imágenes y se procesan según corresponda. Si se encuentra la carpeta del día actual, se llaman a las funciones process_images() y create_video() para cada carpeta, generando el video y registrando su nombre en video_to_upload.txt. Por último, se realizan tareas de limpieza de archivos temporales mediante clean_up().

4.1 Procesamiento de archivos (enviar.py)

El script enviar.py tiene como función principal gestionar la transferencia de archivos a un servidor FTP, llevándose a cabo varias tareas clave:

Carga de configuraciones: Al inicio del script, se lee la configuración desde un archivo externo llamado config.cfg. Este archivo contiene información esencial, como la dirección del servidor FTP, las credenciales necesarias para el acceso y el directorio en el que se almacenarán los archivos en el servidor. Esta etapa permite que el script se adapte fácilmente a distintos entornos sin requerir modificaciones en el código.

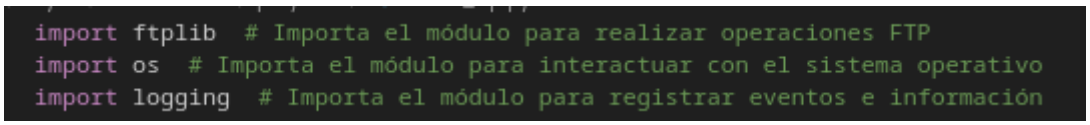
Envío de archivos: La función central del script es la transferencia de archivos a través de FTP. Para ello, verifica la existencia de cada archivo que se quiere subir y organiza su carga en el directorio remoto especificado. Si el directorio no está presente, el script intenta crearlo, asegurando así que los archivos se almacenen correctamente en la estructura del servidor.

Eliminación de archivos locales: Después de que un archivo se ha cargado con éxito, el script procede a eliminar el archivo local. Esto contribuye a mantener el espacio de almacenamiento en el sistema local libre de archivos innecesarios, facilitando una gestión más eficiente de los mismos.

Registro de eventos: Durante su ejecución, el script lleva un registro de eventos significativos, como los resultados de las operaciones de carga y eliminación de archivos. Este registro es fundamental para detectar errores y realizar la depuración, ya que permite identificar rápidamente cualquier problema que pueda surgir.

En este script, utilizamos varias librerías que son cruciales para realizar las operaciones de envío de archivos:

Figura 13. Librerías utilizadas en el script enviar.py



Elaborado por: Bryan Gaibor y Franklin García.

- **os**: Este módulo nos permite manejar rutas y archivos en el sistema, facilitando la gestión de directorios y la eliminación de archivos temporales.
- **ftplib**: Es la herramienta principal que usamos para establecer conexiones y transferir archivos mediante el protocolo FTP, lo que nos facilita la interacción con el servidor remoto.
- **logging**: Proporciona un sistema robusto para registrar eventos del programa, siendo esencial para la depuración y el monitoreo de errores a lo largo de la ejecución del script. Gracias a esta librería, podemos

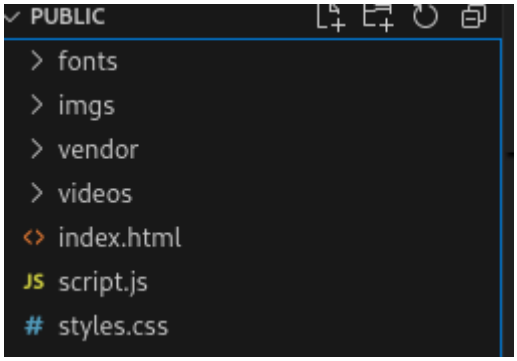
guardar registros en un archivo y, al mismo tiempo, mostrarlos en la consola.

Una vez que hemos abordado los elementos clave del proyecto, es momento de concentrarnos en las funciones principales del código. En esta sección, analizaremos cómo se habilitan estas funciones y cómo están estructuradas para cumplir con los objetivos del script. Este análisis nos permitirá comprender mejor la lógica de la implementación y apreciar la conexión entre las diversas secciones del código.

- **read_config()**: Esta función se encarga de leer la configuración desde el archivo config.cfg. Abre el archivo y procesa cada línea, ignorando las que están vacías o son comentarios. Los pares clave-valor que se extraen se almacenan en un diccionario, que será utilizado a lo largo del programa para acceder a las configuraciones necesarias.
- **send_file_via_ftp(file_path, remote_folder)**: Esta función gestiona el envío de un archivo al servidor FTP. Se conecta al servidor, inicia sesión utilizando las credenciales obtenidas de la configuración y verifica si el directorio remoto existe, creándolo si es necesario. Luego, sube el archivo especificado al directorio remoto y registra si la operación fue exitosa. Si se presenta algún error, se documenta para su seguimiento posterior.
- **main()**: En esta función se controla el flujo principal del programa. Se abre el archivo que contiene la lista de videos a subir y se itera sobre cada entrada. Para cada archivo, se intenta enviarlo mediante la función send_file_via_ftp(). Si la carga se realiza con éxito, el archivo local se elimina para liberar espacio en el sistema. Finalmente, se verifica si todas las operaciones se llevaron a cabo correctamente y se gestiona la eliminación del archivo que contenía la lista de videos.

Estructura del código.

Figura 14 Estructura del código.



Elaborado por: Bryan Gaibor y Franklin García.

Descripción de los Archivos

- **css**: Este archivo alberga la hoja de estilo CSS que determinan cómo se verá la interfaz de usuario. Aquí se configuran diversos aspectos del diseño, como colores, tipos de letra y la disposición de los elementos, todo con el objetivo de crear una experiencia visual atractiva y armónica.
- Script: Se encuentra el script que gestionan la lógica y la interactividad de la aplicación. Este archivo permite que los usuarios interactúen con la interfaz.
- img: Esta carpeta se utiliza para almacenar todas las imágenes que se emplean en la aplicación, tales como iconos, logotipos y otros gráficos que enriquecen la presentación visual.
- videos: Aquí se recopilan los archivos de video que complementan la aplicación. Aquí se incluye los videos, que facilite a los usuarios la comprensión del funcionamiento y las características del proyecto.
- index.html: Este archivo HTML principal funciona como la entrada a la aplicación. Es responsable de cargar la interfaz de usuario y establecer la estructura fundamental de la página, además de vincular los recursos de estilo y los scripts necesarios para su operación.

Para visualizar la estructura completa de la aplicación web revisar el manual técnico véase en anexo.

4.5 Pruebas

4.5.1 Análisis de requisitos

Requisitos de la aplicación:

Tabla 21 Análisis de requisitos

Características	**Requisitos**
1. Selección de Canal y Fecha	Tener conexión a internet
La aplicación web deberá permitir a los usuarios seleccionar el canal (banda espectral) y la fecha.	
2. Visualización de animaciones	Tener conexión a internet. La aplicación web deberá permitir la visualización interactiva de animaciones generadas.
La aplicación web deberá permitir la visualización interactiva de animaciones generadas.	

Elaborado por: Bryan Gaibor y Franklin García

4.5.2 Planificación de pruebas

1. Elaboración de la ficha de observación.
2. Realización de las pruebas.
3. Registro de los resultados obtenidos en las pruebas.

- **Inicio:** 14 de septiembre
- **Finalización:** 20 de septiembre

Actividades de Pruebas:

Tabla 22 Cronograma de pruebas

Actividad	**Día 1**	**Día 2**	**Día 3**	**Día 4**	**Día 5**
Elaboración de la ficha de observación	X				
Realización de la prueba		X	X	X	X
Registro de los resultados de las pruebas	X	X	X	X	X

Elaborado por: Bryan Gaibor y Franklin García

4.5.3 Diseño y desarrollo de casos de prueba

Tabla 23 Caso de Prueba 01: Selección de Canal y Fecha

Elemento	**Descripción**
Característica	La página web deberá permitir a los usuarios seleccionar el canal (banda espectral del satélite GOES-16) y la fecha desde la interfaz.
Requisitos	Tener conexión a internet.
Pasos a seguir	1. Acceda a la aplicación web en el sitio https://www.grintec.org/. 2. Verifique que el dispositivo esté conectado a una red de internet estable. 3. Seleccione el canal y la fecha desde la interfaz.
Resultado esperado	El sistema debe permitir la selección de un canal y una fecha, registrando correctamente los datos ingresados por el usuario.
Criterios de Éxito	El usuario puede seleccionar el canal y la fecha sin errores, y la información queda registrada para su uso posterior.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 24 Caso de Prueba 02: Visualización de Animaciones

Elemento	**Descripción**
Característica	La aplicación web deberá permitir la visualización de animaciones generadas a partir de imágenes satelitales con capas geográficas interactivas.
Requisitos	Tenga conexión a Internet y que los videos sean procesados y cargados en el servidor.
Pasos a seguir	1. Acceda a la aplicación web en el sitio https://www.grintec.org/. 2. Navegar a la sección de visualización de animaciones. 3. Iniciar la reproducción de una animación generada.
Resultado esperado	Las animaciones deben reproducirse correctamente, permitiendo pausar, adelantar, retroceder, hacer zoom y desplazarse por la interfaz sin problemas.
Criterios de Éxito	Las animaciones se reproducen sin errores, y todas las funciones interactivas están operativas con conexión a internet.

Elaborado por: Bryan Gaibor y Franklin García

4.5.4 Configuración del entorno de prueba

Para llevar a cabo las pruebas del proyecto, es necesario contar con los siguientes entornos:

- **Sistema Operativo:** Linux - Debian o Windows.

Además, se recomienda utilizar los navegadores más comunes para garantizar la compatibilidad y el correcto funcionamiento de la aplicación en diversas plataformas

4.5.5 Ejecución de la prueba

Tabla 25 Ficha de prueba de funcionalidad

Elemento	**Descripción**					
Objetivo	Probar la funcionalidad del producto					
Producto	"Visor Grintec"					
Fecha	17/09/2024					
Observador	Dr. Henry Vallejo					
Calificación	1 = no cumple, 5 = cumple satisfactoriamente					
Caso de prueba	**1**	**2**	**3**	**4**	**5**	**Observaciones**
Caso de prueba 01					**X**	La selección de canal y fecha funciona correctamente desde la interfaz.
Caso de prueba 02					**X**	Las animaciones se reproducen con fluidez, permitiendo pausar, adelantar y hacer zoom.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 26 Ficha - Evaluación de Rendimiento

Elemento	**Descripción**
Objetivo	Evaluar el rendimiento del producto
Producto	"Visor Grintec"
Fecha	17/09/2024
Observador	Dr. Henry Vallejo
Calificación	1 = muy lento, 5 = excelente

Caso de Prueba	**1**	**2**	**3**	**4**	**5**	**Observaciones**
Caso de prueba 01: Tiempo de selección de canal y fecha					X	La selección de canal y fecha ocurre en un tiempo razonable, con respuesta eficiente.
Caso de prueba 02: Tiempo de reproducción de animaciones				X		La reproducción de animaciones es fluida, con ligeras demoras en la inicialización.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 27 Ficha - Evaluación de Usabilidad

Elemento	**Descripción**
Objetivo	Evaluar la usabilidad del producto
Producto	"Visor Grintec"
Fecha	19/09/2024
Observador	Dr. Henry Vallejo
Calificación	1 = muy deficiente, 5 = excelente

Caso de prueba	**1**	**2**	**3**	**4**	**5**	**Observaciones**
Caso de prueba 01: Visualización de Información					X	La información se visualiza correctamente en la página web, con buena claridad y accesibilidad.
Caso de prueba 02: Reproducción de Videos Generados					X	Los videos generados se reproducen sin interrupciones y los controles responden de forma efectiva.

Elaborado por: Bryan Gaibor y Franklin García

Pruebas de Caja Negra

Las pruebas de caja negra se centran en evaluar las funcionalidades del sistema basado en los requerimientos establecidos, sin considerar cómo está construido internamente. Estas pruebas permiten verificar que el sistema cumple con los requisitos funcionales especificados en los casos de uso.

Se realizo pruebas exclusivamente para los siguientes requerimientos del sistema:

Visualización de información en tiempo real (RF01).

Reproducción fluida de videos generados y funcionamiento adecuado de los controles (RF02).

Tabla 28 Pruebas de Caja Negra

Caso de Prueba	Descripción	Entradas	Resultado Esperado	Resultado Obtenido	Estado
01: Visualización de Información	Verificar que la información en tiempo real se visualice de forma clara y actualizada en la página web.	Acceso a la página principal	La información debe ser visible, clara, y actualizarse automáticamente sin errores.	La información se visualizó claramente y se actualizó sin fallos.	**Aprobado**
02: Reproducción de Videos	Comprobar que los videos generados se reproduzcan sin interrupciones y los controles funcionen correctamente.	Seleccionar un video y usar los controles	Los videos deben reproducirse de manera fluida, con controles de reproducción operativos.	Reproducción fluida y controles funcionales.	**Aprobado**

Elaborado por: Bryan Gaibor y Franklin García

Pruebas de Caja Blanca

Las pruebas de caja blanca se centran en evaluar el funcionamiento interno del sistema, revisando la lógica del código, las estructuras y los flujos. En este caso, se analizarán los scripts y procesos que soportan los requerimientos clave del sistema para garantizar su correcto funcionamiento.

Se realizarán pruebas exclusivamente para los siguientes requerimientos del sistema:

Visualización de información en tiempo real (flujos y actualización de datos).

Reproducción fluida de videos generados y funcionamiento de controles (lógica de reproducción y manejo de eventos).

Tabla 29 Pruebas de Caja Blanca

Caso de Prueba	Descripción	Entradas	Proceso Interno Evaluado	Resultado Esperado	Resultado Obtenido	Estado
01: Visualización de Información	Verificar que el script actualice correctamente la información en tiempo real.	Datos enviados por sensores remotos	Lógica de actualización de datos y su reflejo en la página web.	Los datos se actualizan en el servidor y se reflejan en tiempo real en la interfaz.	Los datos se actualizaron correctamente.	**Aprobado**
02: Reproducción de Videos	Comprobar que la lógica de reproducción de videos sea eficiente y los controles se procesen correctamente.	Reproducción de un video con controles (play, pause, etc.)	Manejo de eventos para controles y bucles de reproducción.	La reproducción es fluida, y los eventos se procesan sin errores ni retrasos.	Los eventos y la reproducción funcionaron como se esperaba.	**Aprobado**

Elaborado por: Bryan Gaibor y Franklin García

4.5.6 Prueba de cierre

Tabla 30 Prueba de Cierre para el Proyecto "Visor Grintec"

Prueba de cierre del proyecto	
Objetivo	Verificar que todos los aspectos del proyecto estén completos y en conformidad para el cierre
Producto	"Visor Grintec"
Fecha	20/09/2024
Observador	Dr. Henry Vallejo

Aspecto Evaluado	**1**	**2**	**3**	**4**	**5**	**Observaciones**
1. Cumplimiento de Requisitos Funcionales					X	Todos los requisitos funcionales han sido cumplidos satisfactoriamente.
2. Documentación Completa					X	La documentación del proyecto está completa y bien organizada.
3. Resultados de Pruebas de Usabilidad					X	Las pruebas de usabilidad se completaron con calificaciones de 5, indicando una buena experiencia de usuario.
4. Resultados de Pruebas de Rendimiento					X	El rendimiento del sistema cumple con los tiempos esperados, con calificaciones de 5 en todas las pruebas.
5. Cumplimiento de Requisitos de Seguridad					X	Se han cumplido todos los requisitos de seguridad necesarios para el proyecto.
6. Pruebas de Funcionalidad Completas					X	Todos los casos de prueba funcionales fueron completados y evaluados satisfactoriamente.
7. Verificación de la Implementación de Características					X	Las características del producto se han implementado según lo especificado.
8. Verificación de Entregables					X	Todos los entregables del proyecto han sido entregados y revisados.
9. Satisfacción del Cliente o Usuario Final					X	El cliente o usuario final está satisfecho con el producto final y su funcionalidad.

Elaborado por: Bryan Gaibor y Franklin García

5 CONCLUSIONES

- Este proyecto ha mejorado la presentación de imágenes satelitales al crear animaciones time lapse que permiten un monitoreo detallado de los fenómenos meteorológicos en la provincia de Bolívar, facilitando el análisis de la evolución del clima en tiempo real.
- La aplicación web permite el acceso a datos meteorológicos en tiempo real a través de la plataforma Grintec.org, los usuarios pueden acceder de manera interactiva a videos que muestran la evolución de eventos climáticos, lo que mejora la comprensión y análisis de los datos meteorológicos en la región.
- El uso de scripts en Python y tecnologías como OpenCV y FSRCNN permite la automatización del proceso de creación y carga de videos time lapse en un servidor, optimizando el procesamiento de imágenes para su visualización web.
- La herramienta ayuda a autoridades y agricultores a monitorear en tiempo real las condiciones climáticas, mejorando la toma de decisiones y la planificación ante eventos meteorológicos extremos.
- El proyecto utiliza un modelo de arquitectura modular con Node.js y Express.js, lo que asegura un sistema escalable, de fácil mantenimiento y con una interacción eficiente entre servidor y cliente.
- Esta solución tecnológica mejora la alfabetización climática en Bolívar, brindando acceso a información meteorológica en tiempo real, lo que sensibiliza sobre los fenómenos atmosféricos y sus efectos.
- El proyecto integra técnicas avanzadas de procesamiento y visualización de imágenes satelitales, lo que impulsa la investigación en geomática y expande el conocimiento sobre el monitoreo climático a través de tecnologías espaciales.

6 RECOMENDACIONES.

- Migrar o respaldar el almacenamiento local de videos y datos a una solución en la nube para aumentar la escalabilidad, seguridad y accesibilidad de los archivos generados, garantizando la disponibilidad de los videos incluso si ocurre un fallo en el sistema local.
- Implementar un sistema de vigilancia continua del estado del hardware (como la Raspberry Pi 5) y del software utilizado en el procesamiento y transmisión de datos. Esto ayudaría a prevenir fallos mediante mantenimiento predictivo, asegurando la continuidad del servicio.
- Evaluar la posibilidad de abrir el sistema mediante interfaces de programación (APIs) para que otras entidades o investigadores puedan acceder a los datos generados. Esto fomentaría la colaboración y la reutilización de los resultados.

7 BIBLIOGRAFÍA

16_TEMP_CP_2.png (3306×2340). (n.d.). Retrieved May 20, 2024, from https://www.adaptacioncc.com/sites/default/files/2021-05/16_TEMP_CP_2.png

Arquitectura Cliente Servidor | Diagramas / Diseños aportados por usuarios de Visual Paradigm. (n.d.). Retrieved July 15, 2024, from https://online.visual-paradigm.com/es/community/share/arquitectura-cliente-servidor-v0oazbv65

Asamblea Nacional del Ecuador. (2016). *Codigo-Organico-de-la-Economia-Social-de-los-Conocimientos-Creatividad-e-Innovacion*.

Asamblea Nacional del Ecuador. (2022). *LEY ORGÁNICA DE PROTECCIÓN DE DATOS PERSONALES*. www.lexis.com.ec

Banda Pancromática Y El Uso En Las Imágenes Satelitales. (n.d.). Retrieved May 19, 2024, from https://eos.com/es/make-an-analysis/panchromatic/

Cámaras Hiperespectrales. (n.d.). Retrieved May 19, 2024, from https://bcnvision.es/blog-vision-artificial/camaras-hiperespectrales/

Carlos Lobo. (2022). *Los estándares de calidad del software más importantes - Blog de hiberus*. https://www.hiberus.com/crecemos-contigo/los-estandares-de-calidad-del-software-mas-importantes/

Cesar, J., Becerra, A., Elena, C., & Vanegas, D. (2018). Propuesta de un método para desarrollar Sistemas de Información Geográfica a partir de la metodología de desarrollo ágil - SCRUM. *Cuaderno Activa*, *10*, 29–41. https://doi.org/10.53995/20278101.490

Charles R. Severance. (2020). *Python para todos Explorando la información con Python 3 Charles R. Severance*.

Christian Andres Dávila Vilca. (2021). *ACTUALIZACIÓN E IMPLEMENTACIÓN DEL SISTEMA DE RECEPCIÓN SATELITAL GOES-16 PARA EL PRONÓSTICO DEL TIEMPO ATMOSFÉRICO*.

CHRISTIAN ANDRES DÁVILA VILCA. (2021). *ACTUALIZACIÓN E IMPLEMENTACIÓN DEL SISTEMA DE RECEPCIÓN SATELITAL GOES-16 PARA EL PRONÓSTICO DEL TIEMPO ATMOSFÉRICO*.

crontab (5) - Linux manual page. (n.d.). Retrieved September 15, 2024, from https://www.man7.org/linux/man-pages/man5/crontab.5.html

Danielle Losos. (2021). *Beginner's Guide to GOES-R Series Data Document Change Record Version ersion ersion Date Slides Affected Description*.

Debian. (2023). *Debian -- Acerca de Debian*. https://www.debian.org/intro/about.es.html

Folgueiras Bertomeu, P. (2016). La entrevista. *Documents de Treball / Informes (Mètodes d'Investigació i Diagnòstic En Educació)*. https://diposit.ub.edu/dspace/handle/2445/99003

Guzmán-Álvarez, J. A., González-Zuñiga, M., Sandoval Fernandez, J. A., & Calvo-Alvarado, J. C. (2022). *Uso de sensores remotos en la agricultura: aplicaciones en el cultivo del banano*. https://doi.org/10.15517/am.v33i3.48279

Hat Red. (2022, July 19). *¿Qué es la metodología ágil?* https://www.redhat.com/es/topics/devops/what-is-agile-methodology

Ian Sommerville. (2015). *Ingenieria del Software 7ma. Ed.*

LEY DEL INSTITUTO DE METEOROLOGÍA E HIDROLOGÍA, INAMHI. (n.d.). www.lexis.com.ec

mod_proxy - Apache HTTP Server Version 2.4. (n.d.). Retrieved September 15, 2024, from https://httpd.apache.org/docs/2.4/mod/mod_proxy.html

Node.js — About Node.js®. (n.d.). Retrieved September 15, 2024, from https://nodejs.org/en/about

Orlando H. Álvarez Hernández, Thuesman E. Montaño Peralta, & Jorge L. Maldonado Correa. (2018). *COMPENDIO DE APUNTES SOBRE ENERGÍA RENOVABLE Y CLIMA DE LA REGIÓN SUR DE ECUADOR*.

PM2 - Overview | PM2 Plus Documentation. (n.d.). Retrieved September 15, 2024, from https://pm2.io/docs/plus/overview/

¿Qué es el modelo cliente-servidor? Pros y contras - IONOS MX. (n.d.). Retrieved May 26, 2024, from https://www.ionos.mx/digitalguide/servidores/know-how/modelo-cliente-servidor/

Raspberry Pi Documentation. (n.d.). Retrieved May 21, 2024, from https://www.raspberrypi.com/documentation/

Schmit, T. J., Lindstrom, S. S., Gerth, J. J., & Gunshor, M. M. (2018). Applications of the 16 spectral bands on the Advanced Baseline Imager (ABI). *Journal of Operational Meteorology*, *06*(04), 33–46. https://doi.org/10.15191/nwajom.2018.0604

Schwaber Ken, & Sutherland Jeff. (2020). *La Guía de Scrum*.

Servicios FTP (Servicios en red) - Joaquín Andreu - Google Libros. (n.d.). Retrieved May 26, 2024, from https://books.google.com.mx/books?hl=es&lr=&id=JsfTAwAAQBAJ&oi=fnd&pg=PA79&dq=FTP+QUE+ES&ots=0GIjhumq0s&sig=4-srGXcgtdR04zlcvRVtm58U6TU#v=onepage&q=FTP%20QUE%20ES&f=false

Time lapse: qué es y cómo hacerlo | Emilio Simón. (n.d.). Retrieved May 21, 2024, from https://emiliosimon.com/time-lapse/

Visual Studio Code. (n.d.). *Documentation for Visual Studio Code*. Retrieved May 15, 2024, from https://code.visualstudio.com/docs

8 ANEXOS

1. CRONOGRAMA DE ACTIVIDADES Y PRESUPUESTO EJECUTADO

- Cronograma de actividades



Etapa 1: Recolección y Procesamiento de Imágenes Satelitales	15 días	lun 3/6/24	vie 21/6/24
Entrevista	5 días	lun 3/6/24	vie 7/6/24
Extracción de Validación de la calidad y integridad de las imágenes descargadas.mágene	10 días	lun 10/6/24	vie 21/6/24
Etapa 2: Generación de Animaciones	45 días	lun 24/6/24	vie 23/8/24
Desarrollo de la aplicación para generar animaciones.	15 días	lun 24/6/24	vie 12/7/24
Integración de la aplicación para generar animaciones.	15 días	lun 15/7/24	vie 2/8/24
Integración del cliente FTP con la aplicación de generación de animaciones.	15 días	lun 5/8/24	vie 23/8/24
Etapa 3: Desarrollo del Módulo Web	26 días	lun 26/8/24	lun 30/9/24
Implementación del visor de a nimaciones en la web grintec.org.	8 días	lun 26/8/24	mié 4/9/24
Desarrollo de la funcionalidad para reproducir animaciones y navegar entre diferentes fechas y horas.	8 días	jue 5/9/24	lun 16/9/24
Pruebas para asegurar el correcto funcionamiento de la web y de la aplicación	8 días	mar 17/9/24	jue 26/9/24
Entrega del producto Software	2 días	vie 27/9/24	lun 30/9/24

- Presupuesto Ejecutado

Tabla. Presupuesto

N	Recurso	Cantidad	Precio Unitario (USD)	Total (USD)
1	Computadora Portátil	2	$ 1,000.00	$ 2,000.00
2	Servicio de Internet (por mes)	4	$ 35.00	$ 140.00
3	Impresiones	400	$ 0.25	$ 100.00
4	Tinta	4	$ 17.50	$ 70.00
5	Impresora	1	$ 200.00	$ 200.00
6	Resma de papel	2	$ 5.50	$ 11.00
7	Otros gastos operativos	2	$ 100.00	$ 200.00
8	Movilidad	6	$ 10.00	$ 60.00
TOTAL				$ 2,781.00

2. CARTA DE ACEPTACIÓN DE LA ORGANIZACIÓN DONDE SE REALIZÓ EL TRABAJO DE INTEGRACIÓN CURRICULAR.

UEB
UNIVERSIDAD ESTATAL DE BOLIVAR

FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA

Señor

Bryan Gaibor

ESTUDIANTE DE LA UNIVERSIDAD ESTATAL DE BOLIVAR

Presente. -

De mi consideración

Reciba un cordial y atento saludo

Yo, **Dr. Henry Fernando Vallejo Ballesteros,** como **director del proyecto de investigación Análisis de Datos De la red de sensores remotos para el monitoreo del estado del tiempo en los 3 pisos climáticos de la Provincia Bolívar en los periodos 2022 al 2024. Primera Etapa**, me permito autorizar que como parte del proyecto de investigación se desarrolle ejecute el trabajo de integración curricular, denominado: **"TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16".** Se va a dar el acceso a los servidores y facilidades para el cumplimiento de los objetivos.

Agradeciendo por la atención a la presente, me suscribo

Guaranda, 23 de mayo del 2024

Atentamente,

Dr. Henry Fernando Vallejo Ballesteros

Dirección: Av. Ernesto Che Guevara y Gabriel Secaira
Guaranda-Ecuador
Teléfono: (593) 3220 6059
www.ueb.edu.ec

UEB
UNIVERSIDAD ESTATAL DE BOLIVAR

FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA

Señor

Franklin García

ESTUDIANTE DE LA UNIVERSIDAD ESTATAL DE BOLIVAR

Presente. -

De mi consideración

Reciba un cordial y atento saludo

Yo, **Dr. Henry Fernando Vallejo Ballesteros,** como **director del proyecto de investigación Análisis de Datos De la red de sensores remotos para el monitoreo del estado del tiempo en los 3 pisos climáticos de la Provincia Bolívar en los periodos 2022 al 2024. Primera Etapa**, me permito autorizar que como parte del proyecto de investigación se desarrolle ejecute el trabajo de integración curricular, denominado: **"TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16"**. Se va a dar el acceso a los servidores y facilidades para el cumplimiento de los objetivos.

Agradeciendo por la atención a la presente, me suscribo

Guaranda, 23 de mayo del 2024

Atentamente,

Dr. Henry Fernando Vallejo Ballesteros

Dirección: Av. Ernesto Che Guevara y Gabriel Secaira
Guaranda-Ecuador
Teléfono: (593) 3220 6059
www.ueb.edu.ec

3. Instrumentos de recopilación de datos

Entrevista para la Recolección de Datos

Proyecto: Time Lapse a partir de Imágenes Satelitales en Tiempo Real del GOES-16, en la Provincia Bolívar

Entrevistadores Sr. Franklin García y Sr. Bryan Hernán Gaibor Naranjo.

Entrevistado: Dr. Henry Vallejo

Sección 1: Identificación de Requerimientos

1. **¿Cuáles son las principales necesidades que el sistema debe abordar en términos de procesamiento de imágenes satelitales?**
   El sistema debe procesar automáticamente las imágenes satelitales descargadas del GOES-16, recortarlas para enfocar exclusivamente la provincia de Bolívar, superponer capas geográficas que muestren divisiones administrativas, y generar animaciones (time lapses) que sean claras y útiles para el análisis climático.

2. **¿Qué funcionalidades considera esenciales para el sistema (por ejemplo, automatización, interfaz web, capacidad de carga automática)?**

   - Automatización del recorte y procesamiento de imágenes mediante crontab.
   - Generación de animaciones en formato video a partir de las imágenes procesadas.
   - Subida automática de los videos generados al servidor mediante FTP.
   - Una interfaz web accesible para visualizar las animaciones de forma interactiva.

3. **¿Qué datos o información adicional, además de las imágenes satelitales, sería útil incluir en el sistema?**
   Sería valioso incluir barras de colores para poder leer y distinguir de mejor forma el espectro de colores que muestra la imagen

4. **¿Cuáles son las expectativas de los usuarios finales en términos de accesibilidad y uso del sistema?**
   Se espera que el sistema sea fácil de usar incluso para usuarios con conocimientos técnicos básicos. La interfaz debe ser intuitiva, responsiva, y funcional en múltiples dispositivos, incluidos smartphones y tablets.

5. **¿Qué características debe tener la interfaz web para facilitar la navegación y el acceso a las animaciones?**

    - Un selector de fechas y canales para elegir qué animaciones visualizar.
    - Controles de reproducción (pausar, adelantar, retroceder, pantalla completa).
    - Una estructura clara con menús y botones intuitivos.
    - Compatibilidad con navegadores modernos como Chrome, Firefox y Edge.

6. **¿Qué tan importante considera la automatización de tareas como el recorte de imágenes y la carga al servidor mediante crontab?**
   Es crucial, ya que elimina la necesidad de intervención manual y asegura la actualización continua del sistema.

7. **¿Existen estándares técnicos o restricciones (como formatos de archivo o herramientas específicas) que el sistema deba cumplir?**
   El sistema debe cumplir con estándares web como HTML5 para la interfaz y utilizar formatos comunes como PNG para imágenes y MP4 para videos.

8. **¿Qué desafíos anticipa en la integración de las herramientas utilizadas, como Python, OpenCV y crontab, para garantizar un sistema fluido?**
   Los desafíos incluyen la gestión de recursos en la Raspberry Pi, como memoria y almacenamiento, y asegurar que los scripts programados mediante crontab no se vean afectados por interrupciones en la red o fallos en el hardware.

9. **¿Qué métricas o indicadores serían útiles para medir el éxito del sistema (por ejemplo, tiempo de procesamiento, calidad de las animaciones, usabilidad de la interfaz)?**

    - Tiempo de procesamiento desde la descarga de imágenes hasta la generación del video (idealmente menos de 5 minutos).
    - Calidad visual de las animaciones, incluyendo nitidez y superposición precisa de capas.
    - Tasa de accesos y satisfacción de los usuarios en la interfaz web.

10. **¿Qué mejoras futuras imagina para este sistema una vez que esté en operación?**

    - Integración con más fuentes de datos climáticos para ofrecer análisis más detallados.
    - Implementación de alertas automáticas basadas en condiciones climáticas específicas.

Firma del Entrevistado:
Dr. Henry Fernando Vallejo Ballesteros

Firma del Entrevistador:
Sr. Bryan Hernán Gaibor Naranjo

Firma del Entrevistador:

Sr. Franklin Sebastián García Villares

4.MANUAL DE USUARIO

INGENIERÍA EN SOFTWARE

RSR-MET



VISOR GRINTEC

Manual de Usuario

PROYECTO: TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.

Versión 1.0

ELABORADO POR:
Bryan Hernán Gaibor Naranjo
Franklin Sebastián García Villares

HOJA DE CONTROL

Organismo	Universidad Estatal de Bolívar		
Proyecto	TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.		
Entregable	Manual de Usuario		
Autor	- Bryan Gaibor - Franklin García		
Versión/Edición	01	**Fecha Versión**	10/09/2024
Aprobado por	- Dr. Henry Vallejo	**Fecha Aprobación**	14/09/2024
		Nº Total de Páginas	10

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
01	Versión inicial	Bryan Gaibor Franklin García	14/09/2024

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
Bryan Hernán Gaibor Naranjo
Franklin Sebastián García Villares

CONTENIDO

1 DESCRIPCIÓN DEL SISTEMA 6
1.1 Objetivo 6
1.2 Alcance 6
1.3 Funcionalidad 6
2 MAPA DEL SISTEMA 8
2.1 Modelo Lógico 8
2.2 Navegación 9
3 DESCRIPCIÓN DEL SISTEMA 10
3.1.1 Pantalla 11
3.1.2 Mensajes de error 14

CONTENIDO

figura 1. Patrón de navegación 9
figura 2. Página de inicio del sitio grintec 11
figura 3. Visor Grintec 11
figura 4. Selección de canales 12
Figura 5. Historial de videos 13
Figura 6 Control del reproductor 13

1 DESCRIPCIÓN DEL SISTEMA

1.1 Objetivo

Desarrollar el manual de usuario para la aplicación web “Visor Grintec”, diseñada para la presentación de un visor de time-lapse del clima en la provincia de Bolívar utilizando la plataforma web.

1.2 Alcance

El sistema para generar Time Lapse a partir de imágenes extraídas del GOES-16 proporciona una plataforma de información en línea para la generación y visualización de animaciones de datos climáticos en tiempo real de la provincia de Bolívar, incluyendo:

- **Generación de animaciones**: Procesamiento de imágenes satelitales y superposición de capas para crear videos time lapse.
- **Subida automática:** Carga de los videos generados a un servidor web.
- **Visualización en línea:** Interfaz web que permite la visualización interactiva y ordenada de las animaciones generadas.

1.3 Funcionalidad

Generación de Animaciones:

- Procesamiento de imágenes descargadas del satélite **GOES-16** para generar videos en formato **time lapse**.
- Superposición de capas geográficas que muestran las provincias del Ecuador en las imágenes.
- Compresión y optimización del video generado para mejorar la eficiencia en el almacenamiento y transmisión.

Subida Automática de Videos:

- Los videos generados se suben automáticamente a un servidor FTP, donde se almacenan para su visualización posterior.
- Los videos están organizados en carpetas por canal o banda del satélite GOES-16.

Visualización en Línea:

- Interfaz web interactiva que permite a los usuarios visualizar los videos time lapse.

- Los usuarios pueden seleccionar diferentes canales o bandas del satélite, lo que cambiará el conjunto de videos disponibles para su visualización.
- Navegación cronológica de los videos, puede desplazar entre fechas de los videos del canal que el usuario haya seleccionado.
- Funcionalidad para reproducir las animaciones generadas a partir de imágenes satelitales, con controles para pausar, adelantar o retroceder las animaciones.

2 MAPA DEL SISTEMA

2.1 Modelo Lógico

El sistema "Visor Grintec" es una aplicación web que permite a los usuarios visualizar animaciones en tiempo real capturadas por el satélite GOES-16. Los usuarios acceden a la aplicación a través de un navegador web al sitio https://www.grintec.org, desde el cual pueden interactuar con el visor para explorar diferentes canales y poder visualizar los videos generados.

Generación de Animaciones:

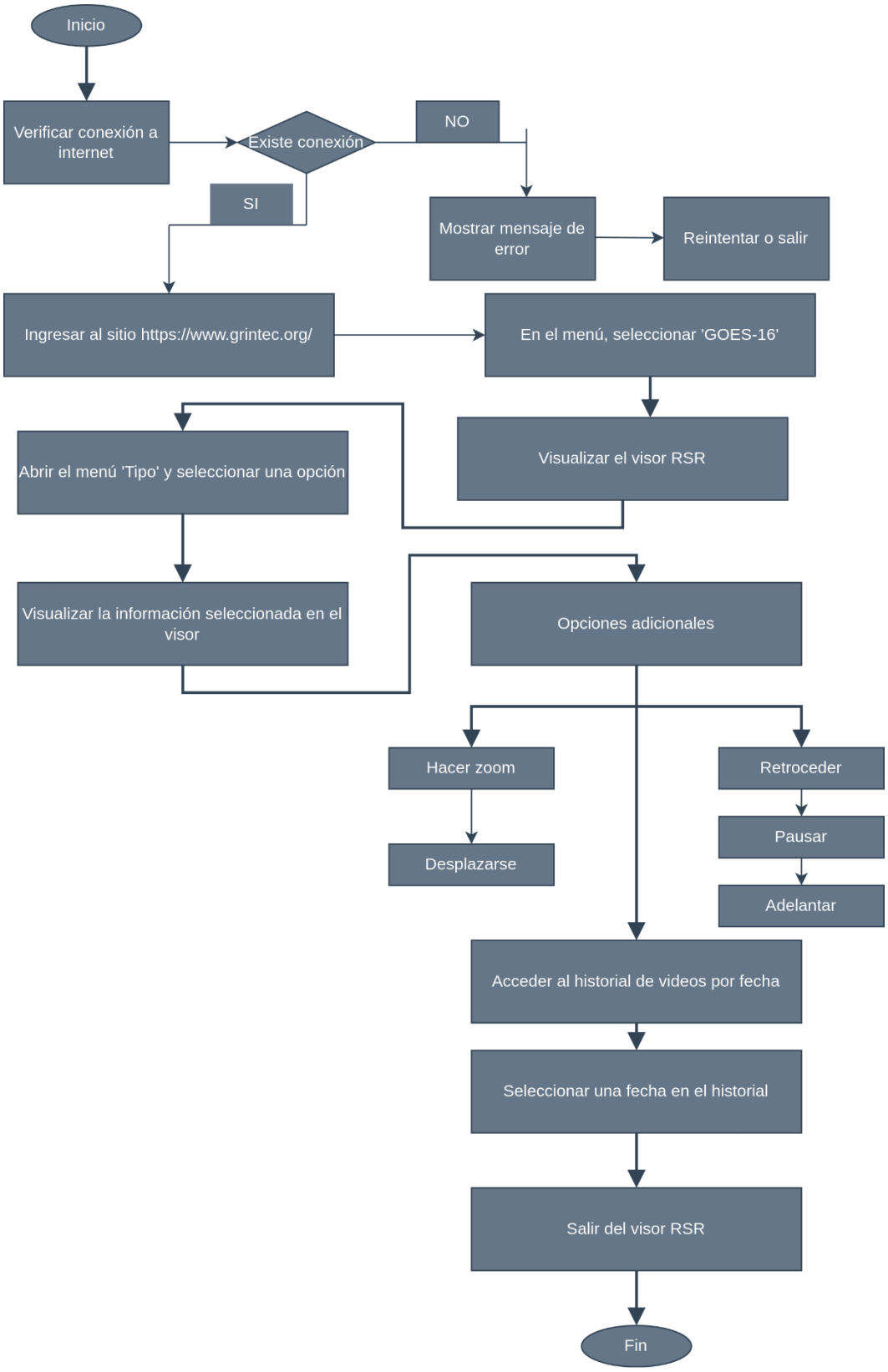
- **Procesamiento de Imágenes:** Convierte las imágenes capturadas del satélite GOES-16 en videos de time-lapse.
- **Subida Automática de Videos:** Envía los videos generados al servidor FTP para su almacenamiento y disponibilidad.

Interfaz de Usuario:

- **Visualización Interactiva:** Permite a los usuarios ver y navegar entre las animaciones de manera interactiva, seleccionando diferentes canales y fechas.

2.2 Navegación

figura 1. Patrón de navegación



Elaborado por: Bryan Gaibor y Franklin Garcia

3 DESCRIPCIÓN DEL SISTEMA

1. Pantalla de Inicio

- **Barra de Navegación:** Incluye enlaces a las principales secciones de la aplicación, como “Inicio”, “tipo/banda”, “Historial”.
- **Imagen de Cabecera:** Un logo de la Universidad Estatal de Bolívar y logo de la nasa con el nombre del canal.
- **Botón de Acceso a Animaciones:** Un botón prominente que lleva a la sección de “Visor Grintec”.

2. Pantalla de Exploración de Animaciones

- **Selector de Canal/Banda:** Permite al usuario elegir el canal o banda del satélite GOES-16.
- **Selector de Fecha:** Un control deslizante o calendario para seleccionar la fecha de las animaciones.
- **Vista de Animaciones:** Muestra una lista o galería de videos disponibles basados en el canal seleccionado.
- **Controles de Reproducción:** Permite pausar, adelantar, retroceder y ajustar el zoom de las animaciones.

3. Pantalla de Historial de Animaciones

- **Lista de Videos Anteriores:** Muestra un historial de videos organizados por fecha y canal.
- **Botón de Reproducción Rápida:** Permite acceder rápidamente a los videos del historial.
- **Opciones de Filtro:** Permite filtrar videos por fecha, canal o banda.

3.1.1 Pantalla

Para ingresar a a la aplicación web “Visor Grintec” acceder al sitio https://www.grintec.org.

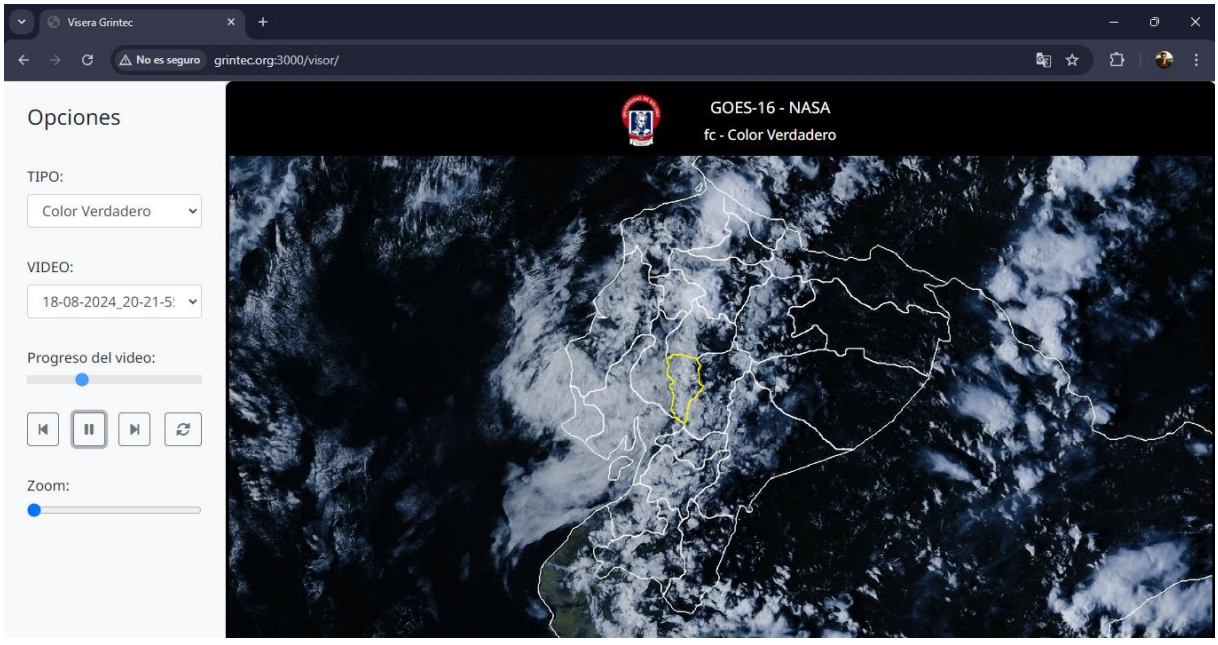
figura 2. Página de inicio del sitio grintec



Elaborado por: Bryan Gaibor y Franklin Garcia

En el menú del sitio nos dirigimos a “GOES-EC” para acceder al visor.

figura 3. Visor Grintec



Elaborado por: Bryan Gaibor y Franklin Garcia

En esta pagina podemos realizar los siguientes pasos:

- En la sección tipo, se elige o selecciona el canal que desea reproducir.
- En el apartado progreso del video se utiliza los controles de reproducción que permite pausar, adelantar, retroceder y ajustar el zoom de las animaciones.
- Permite acceder rápidamente a los videos del historial donde lo pude filtrar por fecha y canal.

figura 4. Selección de canales

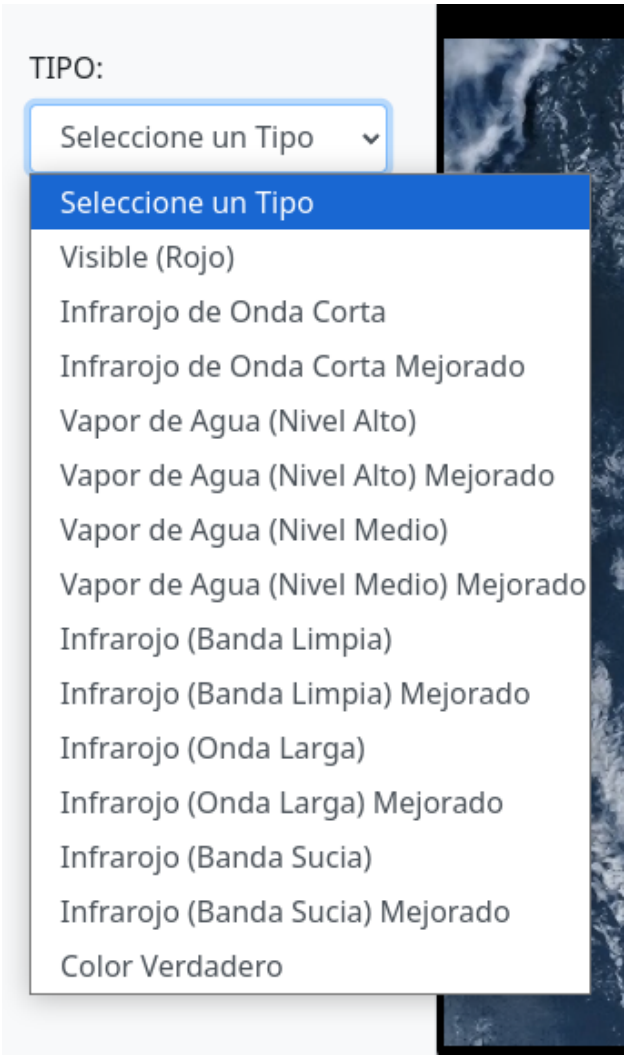


Figura 5. Historial de videos

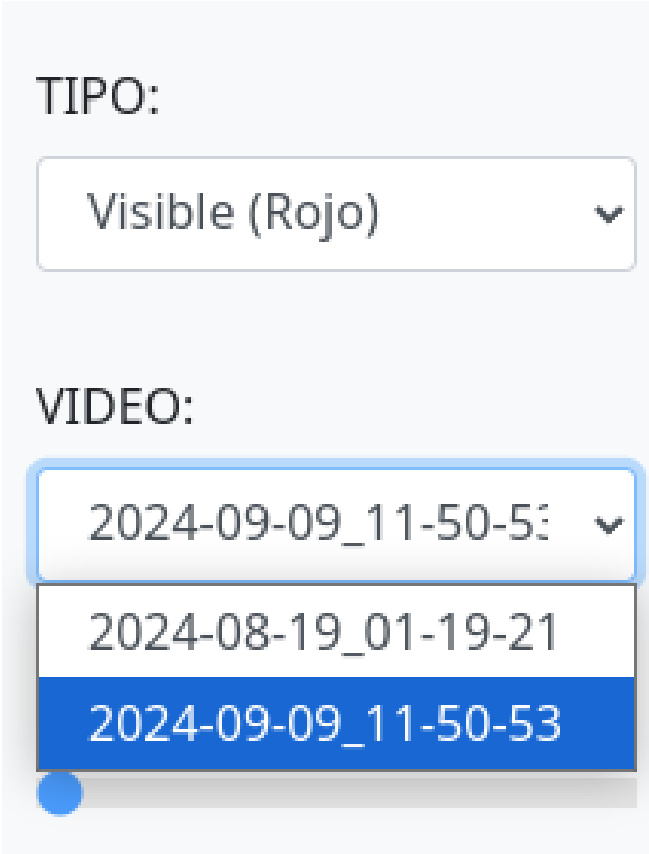
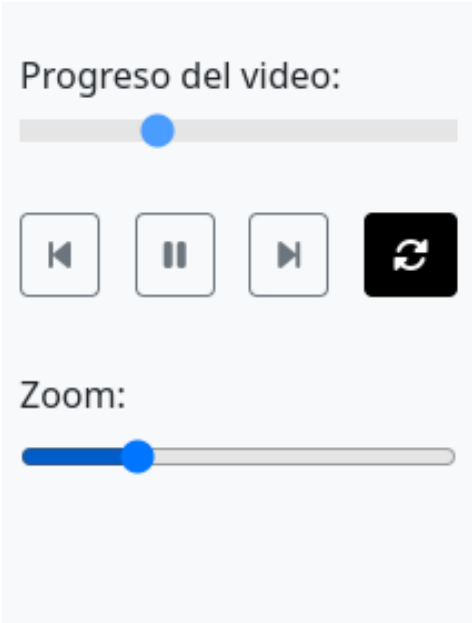


Figura 6 Control del reproductor



3.1.2 Mensajes de error

Mensajes de Error:

- **Error de Carga:** “No se pudo cargar las imágenes. Intente nuevamente más tarde.”
- **Ayuda Contextual:** “Si el video no se carga, verifique su conexión a Internet.”

Mensajes de Error:

- **Error de Carga de Video:** “No se pudo cargar el video. Intente seleccionar otra fecha o canal.”
- **Ayuda Contextual:** “Verifique su conexión a Internet si los videos no se cargan.”

Mensajes de Error:

- **Error de Filtrado:** “No se encontraron videos que coincidan con los criterios seleccionados.”
- **Ayuda Contextual:** “Intente ajustar los filtros o verifique la fecha y canal seleccionados.”

5.

ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

INGENIERÍA EN SOFTWARE

RSR-MET



VISOR GRINTEC

ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

PROYECTO: TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.

Versión 1.0

ELABORADO POR:

Bryan Hernán Gaibor Naranjo

Franklin Sebastián García Villares

Revisión Histórica

Nombre	Fecha	Motivo	Versión

Contenido

1 Introducción 6
1.1.1 Propósito 6
1.2 Alcance 6
1.3 Personal involucrado 6
1.4 Definiciones, acrónimos y abreviaturas 8
1.5 Referencias 10
1.6 Resumen 10
2. Descripción general 10
2.1 Perspectiva del producto 10
2.2 Funcionalidad del producto 10
2.4 Restricciones 11
2.5 Suposiciones 11
3 Requerimientos específicos 12
3.1 Requerimientos funcionales 12
3.2 Requerimientos no funcionales 13
3.2.1 Requerimientos de interfaz 16
3.3 Requerimientos de la interfaz. 18
3.3.1 Interfaces de usuario 18
3.3.2 Interfaces de hardware 19
3.3.3 Interfaces de software 19

Contenido de tablas

Tabla 1. Información del Analista, diseño y programador 6
Tabla 2. Información del Analista, diseñador y programador 7
Tabla 3. Información del usuario 7
Tabla 4. Información del Administrador 7
Tabla 5. Definiciones de acrónimos, siglas y abreviaturas usadas en el documento ERS8
Tabla 6. Referencias 10
Tabla 6 Requerimiento funcional 01: Selección de Canal y Fecha 12
Tabla 7. Requerimiento funcional 02: Visualización de animaciones 12
Tabla 8. Requerimientos no funcionales 01: Funcionalidad 13
Tabla 9. Requerimientos no funcionales 02: Usabilidad. 13
Tabla 10. Requerimientos no funcionales 03: Disponibilidad 14
Tabla 11. Requerimientos no funcionales 04: Compatibilidad 14
Tabla 12. Requerimientos no funcionales 05: Rendimiento 14
Tabla 13. Requerimientos no funcionales 06: Seguridad 15
Tabla 14. Requerimientos no funcionales 07: Generación Automática de Animaciones 15
Tabla 15. Requerimientos no funcionales 08: Carga Automática al Servidor. 15
Tabla 16. Requerimientos no funcionales 09: Automatización del recorte y envió de archivos. 16
Tabla 17. Requerimiento de interfaz 01: Interfaz de usuario. 16
Tabla 18. Requerimiento de interfaz 02: Compatibilidad de navegadores. 17
Tabla 19. Requerimiento de interfaz 03: Interfaz de hardware. 17
Tabla 20. Requerimiento de interfaz 04: Interfaz de software 17

1 Introducción

Este documento es una Especificación de Requisitos Software (ERS) para el sistema de generación y visualización de animaciones a partir de imágenes satelitales del GOES-16 en la provincia de Bolívar. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

1.1.1 Propósito

La aplicación web "Visor Grintec" proporciona un visor que presenta time-lapse para el monitoreo del estado del tiempo de la Provincia Bolívar en tiempo real.

1.2 Alcance

El sistema para generar Time Lapse a partir de imágenes extraídas del GOES-16 proporciona una plataforma de información en línea para la generación y visualización de animaciones de datos climáticos en tiempo real de la provincia de Bolívar, incluyendo:

- **Generación de animaciones**: Procesamiento de imágenes satelitales y superposición de capas para crear videos time lapse.
- **Subida automática:** Carga de los videos generados a un servidor web.
- **Visualización en línea:** Interfaz web que permite la visualización interactiva y ordenada de las animaciones generadas.

1.3 Personal involucrado

Tabla 1. Información del Analista, diseño y programador

Nombre	Bryan Gaibor
Rol	Analista, diseñador y programador
Responsabilidad	Recopilación de datos, diseño y desarrollo de la aplicación web “Visor GRINTEC”.
Información de contacto	brygaibor@mailes.ueb.edu.ec

Elaborado por: Bryan Gaibor y Franklin García

Tabla 2. Información del Analista, diseñador y programador

Nombre	Franklin Garcia
Rol	Analista, diseñador y programador
Responsabilidad	Recopilación de datos, diseño y desarrollo de la aplicación web “Visor GRINTEC”.
Información de contacto	fgarcia@mailes.ueb.edu.ec

Elaborado por: Bryan Gaibor y Franklin García

Tabla 3. Información del usuario

Tipo de usuario	**Usuario**
Características	Personas interesadas en información climática y visualización de animaciones satelitales, con conocimientos básicos en navegación en internet
Actividades	Acceder y visualizar animaciones del clima generadas a partir de imágenes satelitales del GOES-16, interpretar datos climáticos, y enviar sugerencias o consultas a través de la plataforma web.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 4. Información del Administrador

Tipo de usuario	**Administrador**
Características	Personal técnico del proyecto GRINTEC encargado de la gestión y mantenimiento del sistema.
Actividades	Monitorear y asegurar el correcto funcionamiento de los scripts que generan y suben las animaciones, gestionar la infraestructura de hardware y software, y realizar tareas de mantenimiento y actualización del sistema.

Elaborado por: Bryan Gaibor y Franklin García

1.4 Definiciones, acrónimos y abreviaturas

Tabla 5. Definiciones de acrónimos, siglas y abreviaturas usadas en el documento ERS

NOMBRE	DESCRIPCIÓN
GOES-16	Geostationary Operational Environmental Satellite 16, un satélite meteorológico operado por NOAA.
Satélites Meteorológicos	Satélites que llevan instrumentos para observar la Tierra y su atmósfera.
Sensores Remotos	Dispositivos electrónicos diseñados para detectar y medir características del entorno desde una distancia, como temperatura, humedad, presión atmosférica, radiación solar, entre otros, mediante la captación de señales electromagnéticas.
Imágenes Satelitales	Fotografías o datos visuales capturados por satélites artificiales en órbita alrededor de la Tierra
Rpi	Raspberry Pi, una serie de computadoras de placa única de bajo costo.
OpenCV	Open Source Computer Vision Library, una biblioteca de visión artificial y procesamiento de imágenes.
HTTP	Protocolo de Transferencia de Hipertexto, base de la comunicación de datos en la web.
JSON	Notación de Objetos de JavaScript, un formato de intercambio de datos ligero.
HTML	Lenguaje de Marcado de Hipertexto, el estándar para la creación de páginas web.
CSS	Hojas de Estilo en Cascada, utilizadas para diseñar y maquetar páginas web.

JavaScript	Lenguaje de programación utilizado para crear contenido dinámico en las páginas web.
IDE	Entorno de Desarrollo Integrado, una aplicación que proporciona herramientas y servicios integrales para el desarrollo de software.
QA	Aseguramiento de la calidad, actividades que garantizan la calidad del producto final.
ZULU	Término utilizado en aviación, navegación y operaciones militares para referirse al Tiempo Universal Coordinado (UTC)
SDR	Radio Definida por Software utiliza software para realizar funciones de procesamiento de señales de radio, ofreciendo flexibilidad y actualización fácil en comparación con hardware tradicional.
FTP	Protocolo de Transferencia de Archivos, utilizado para la transferencia de archivos entre sistemas.
Express.js	Un framework web para Node.js, diseñado para construir aplicaciones web y APIs de manera eficiente.
Node.js	Entorno de ejecución de JavaScript que permite ejecutar código JavaScript del lado del servidor.
PM2	Administrador de procesos para Node.js que facilita la gestión de aplicaciones en producción.
Apache2	Un servidor web de código abierto que proporciona funcionalidades HTTP.
Crontab	Herramienta de Unix/Linux que permite programar la ejecución de scripts o comandos a intervalos específicos.
Superresolución	Técnica de procesamiento de imágenes utilizada para aumentar la resolución de una imagen más allá de su calidad original.
MoviePy	Biblioteca Python para edición de videos que permite crear, modificar y exportar clips de video.
FSRCNN	Modelo de redes neuronales convolucionales para superresolución rápida, utilizado para mejorar la calidad de las imágenes.
Canal/Banda ABI	Se refiere a las distintas frecuencias del espectro electromagnético que el satélite **GOES-16** utiliza a través del instrumento **ABI (Advanced Baseline Imager)** para capturar diferentes tipos de imágenes climáticas y atmosféricas. Los usuarios del sistema pueden

	seleccionar distintos canales o bandas ABI para ver las imágenes correspondientes.

1.5 Referencias

Tabla 6. Referencias

Título Del Documento	Referencia
Standard IEEE 830-1998	IEEE
IEC 20246-2017	ISO

Elaborado por: Bryan Gaibor y Franklin García

1.6 Resumen

Este documento de Especificación de Requisitos de Software (ERS) está dividido en dos partes. La primera sección proporciona una introducción general al documento y ofrece una visión global de los recursos del sistema especificados. La segunda sección describe la aplicación web, destacando las principales funciones requeridas, así como las restricciones, supuestos y dependencias que podrían afectar el proceso de desarrollo.

2. Descripción general

2.1 Perspectiva del producto

El sistema se compone de dos scripts principales que se ejecutan en una Raspberry Pi 5 y una plataforma web desarrollada con Express.js que se despliega en el servidor www.grintec.org. Los scripts automatizan el procesamiento de imágenes satelitales del GOES-16 y la generación de videos time lapse. Además, gestionan la subida automática de estos videos a un servidor FTP. La plataforma web, basada en Node.js y Express.js, permite a los usuarios visualizar las animaciones de manera interactiva y cronológica a través de un visor web optimizado.

2.2 Funcionalidad del producto

Generación de Animaciones:

- Procesamiento de imágenes descargadas del satélite **GOES-16** para generar videos en formato **time lapse**.
- Superposición de capas geográficas que muestran las provincias del Ecuador en las imágenes.
- Compresión y optimización del video generado para mejorar la eficiencia en el almacenamiento y transmisión.

Subida Automática de Videos:

- Los videos generados se suben automáticamente a un servidor FTP, donde se almacenan para su visualización posterior.
- Los videos están organizados en carpetas por canal o banda del satélite GOES-16.

Visualización en Línea:

- Interfaz web interactiva que permite a los usuarios visualizar los videos time lapse.
- Los usuarios pueden seleccionar diferentes canales o bandas del satélite, lo que cambiará el conjunto de videos disponibles para su visualización.
- Navegación cronológica de los videos, puede desplazar entre fechas de los videos del canal que el usuario haya seleccionado.
- Funcionalidad para reproducir las animaciones generadas a partir de imágenes satelitales, con controles para pausar, adelantar o retroceder las animaciones.

2.4 Restricciones

- **Conectividad:** Requiere conexión a internet para la descarga de imágenes y la subida de videos.
- **Compatibilidad:** La interfaz web debe ser compatible con los navegadores más utilizados, debe soportar HTML5 o superior.

2.5 Suposiciones

- Se supone que se realizara todo lo programado.
- Los usuarios deben tener dispositivos con los navegadores más utilizados para que la aplicación sea compatible.

3 Requerimientos específicos

3.1 Requerimientos funcionales

Tabla 7 Requerimiento funcional 01: Selección de Canal y Fecha

Identificación del Requerimiento	**RF01**
Prioridad	Alta
Nombre del Requerimiento	Selección de Canal y Fecha
Actores	Usuario
Descripción del Requerimiento	Los usuarios podrán seleccionar el canal (banda espectral del satélite GOES-16) y la fecha desde la interfaz web para visualizar las animaciones.
Precondiciones	Tener acceso a Internet y un dispositivo compatible con navegadores web modernos.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 8. Requerimiento funcional 02: Visualización de animaciones

Identificación del Requerimiento	**RF02**
Prioridad	Alta
Nombre del Requerimiento	Visualización de Animaciones
Actores	Usuario

Descripción del Requerimiento	El sistema permitirá visualizar animaciones generadas a partir de imágenes satelitales, con superposición de capas geográficas que muestran las provincias de Ecuador.
Controles	El usuario podrá pausar, adelantar, retroceder y cambiar a pantalla completa.
Precondiciones	Los videos deben estar correctamente procesados y cargados al servidor.

Elaborado por: Bryan Gaibor y Franklin García

3.2 Requerimientos no funcionales

Tabla 9. Requerimientos no funcionales 01: Funcionalidad

Identificación del Requerimiento	**RNF01**
Prioridad	Alta
Nombre del Requerimiento	Funcionalidad
Descripción del Requerimiento	El sistema debe operar correctamente en todo momento, garantizando que las animaciones se generen, procesen y suban sin errores.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 10. Requerimientos no funcionales 02: Usabilidad.

Identificación del Requerimiento	**RNF02**
Prioridad	Alta
Nombre del Requerimiento	Usabilidad
Descripción del Requerimiento	La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo a los usuarios acceder a las animaciones sin necesidad de conocimientos técnicos avanzados.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 11. Requerimientos no funcionales 03: Disponibilidad

Identificación del Requerimiento	**RNF03**
Prioridad	Alta
Nombre del Requerimiento	Disponibilidad
Descripción del Requerimiento	La plataforma debe estar disponible para los usuarios en todo momento, con tiempos de inactividad mínimos.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 12. Requerimientos no funcionales 04: Compatibilidad

Identificación del Requerimiento	**RNF04**
Prioridad	Media
Nombre del Requerimiento	Compatibilidad
Descripción del Requerimiento	La plataforma debe ser compatible con los navegadores web más utilizados, como Google Chrome, Firefox y Microsoft Edge, y debe soportar HTML5.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 13. Requerimientos no funcionales 05: Rendimiento

Identificación del Requerimiento	**RNF05**
Prioridad	Media
Nombre del Requerimiento	Rendimiento
Descripción del Requerimiento	El sistema debe generar y cargar las animaciones en menos de 5 minutos después de recibir las imágenes del satélite.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 14. Requerimientos no funcionales 06: Seguridad

Identificación del Requerimiento	**RNF06**
Prioridad	Alta
Nombre del Requerimiento	Seguridad
Descripción del Requerimiento	Los datos y videos transferidos deben estar protegidos mediante medidas de seguridad, como autenticación en el FTP y cifrado de datos sensibles.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 15. Requerimientos no funcionales 07: Generación Automática de Animaciones

Identificación del Requerimiento	**RNF07**
Prioridad	Alta
Nombre del Requerimiento	Generación Automática de Animaciones
Actores	Sistema
Descripción del Requerimiento	El sistema debe procesar automáticamente las imágenes satelitales del GOES-16 para crear animaciones, incluyendo recorte y superposición de capas geográficas.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 16. Requerimientos no funcionales 08: Carga Automática al Servidor.

Identificación del Requerimiento	**RNF08**
Prioridad	Alta
Nombre del Requerimiento	Carga Automática al Servidor

Actores	Sistema
Descripción del Requerimiento	Los videos generados deben ser cargados automáticamente al servidor web mediante un cliente FTP.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 17. Requerimientos no funcionales 09: Automatización del recorte y envió de archivos.

Identificación del Requerimiento	**RNF09**
Prioridad	Alta
Nombre del Requerimiento	Automatización del Recorte y Envío de Archivos
Actores	Sistema
Descripción del Requerimiento	El sistema debe automatizar el recorte de imágenes satelitales y el envío de los archivos generados al servidor mediante funciones cron en Linux. Estas tareas deben ejecutarse de manera periódica y sin intervención manual.
Precondiciones	El cron debe estar configurado correctamente en el sistema operativo.

Elaborado por: Bryan Gaibor y Franklin García

3.2.1 Requerimientos de interfaz

Tabla 18. Requerimiento de interfaz 01: Interfaz de usuario.

Identificación del Requerimiento	**RI01**
Prioridad	Alta
Nombre del Requerimiento	Interfaz de Usuario
Descripción del Requerimiento	La interfaz web debe ser intuitiva y fácil de navegar, con menús y botones claramente etiquetados, y un diseño responsivo que se

	adapte a diferentes tamaños de pantalla. La interfaz gráfica debe ser agradable a la vista y permitir al usuario identificar rápidamente los componentes del sistema.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 19. Requerimiento de interfaz 02: Compatibilidad de navegadores.

Identificación del Requerimiento	**RI02**
Prioridad	Media
Nombre del Requerimiento	Compatibilidad de Navegadores
Descripción del Requerimiento	La plataforma debe ser compatible con los navegadores web más comunes, incluyendo Google Chrome, Firefox, y Microsoft Edge, para asegurar el acceso universal de los usuarios.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 20. Requerimiento de interfaz 03: Interfaz de hardware.

Identificación del Requerimiento	**RI03**
Prioridad	Alta
Nombre del Requerimiento	Interfaz de Hardware
Descripción del Requerimiento	El sistema debe funcionar en una Raspberry Pi para el procesamiento y generación de las animaciones. La Raspberry Pi debe tener una tarjeta de red, procesador de al menos 1GHz, memoria RAM de al menos 1GB, y almacenamiento de al menos 8GB.

Elaborado por: Bryan Gaibor y Franklin García

Tabla 21. Requerimiento de interfaz 04: Interfaz de software

Identificación del Requerimiento	RI04
Prioridad	Alta
Nombre del Requerimiento	Interfaz de Software
Descripción del Requerimiento	La plataforma debe ser compatible con los sistemas operativos y software necesarios para la Raspberry Pi, incluyendo Raspbian y Python. Además, debe permitir el envío de archivos vía FTP al servidor web donde está alojada la web grintec.org.

Elaborado por: Bryan Gaibor y Franklin García

3.3 Requerimientos de la interfaz.

La interfaz gráfica de la aplicación web "Visor Grintec", que muestra datos en tiempo real del satélite GOES-16 del Ecuador en la provincia Bolivar, debe ser intuitiva y fácil de usar, permitiendo a los usuarios identificar rápidamente los elementos y funcionalidades del sistema. Además, debe utilizar una paleta de colores que resulte agradable a la vista, facilitando su uso prolongado sin causar fatiga visual. La interfaz también debe asegurar la compatibilidad con los navegadores más utilizados, como Google Chrome, Firefox y Microsoft Edge, garantizando una experiencia consistente para todos los usuarios.

3.3.1 Interfaces de usuario

La interfaz de "Visor Grintec" será completamente responsiva, permitiendo su visualización desde cualquier dispositivo, ya sea computadora, tablet o smartphone, independientemente del sistema operativo. Se diseñará para adaptarse automáticamente a diferentes tamaños de pantalla, proporcionando una experiencia de usuario consistente e intuitiva en todas las plataformas. Los componentes de la interfaz, como botones, campos de entrada y etiquetas, estarán organizados de manera que faciliten la interacción y navegación en cualquier dispositivo.

3.3.2 Interfaces de hardware

Para asegurar el rendimiento óptimo de la aplicación web "Visor Grintec", se sugiere utilizar dispositivos con las siguientes especificaciones mínimas de hardware:

- Una conexión a Internet confiable.
- Procesador con una velocidad mínima de 1 GHz.
- Memoria RAM de al menos 1 GB.
- Pantalla con una resolución adecuada para una visualización nítida.

3.3.3 Interfaces de software

- **Navegador web**: Compatible con las versiones actuales de Google Chrome, Firefox y Microsoft Edge.
- **Sistema operativo**: La aplicación es accesible desde cualquier sistema operativo moderno, como Windows, macOS, iOS y Android.
- **Configuración del navegador**: Verificar que JavaScript y cookies estén habilitados para garantizar un funcionamiento correcto.

6. PLAN DE PRUEBAS

INGENIERÍA EN SOFTWARE

RSR-MET



VISOR GRINTEC

PLAN DE PRUEBAS

PROYECTO: TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.

Versión 1.0

ELABORADO POR:

Bryan Hernán Gaibor Naranjo

Franklin Sebastián García Villares

Contenido

Introducción 5
Objetivos 5
Tipos de Pruebas 5
Pruebas Funcionales 6
Pruebas de Rendimiento 7
Pruebas de Usabilidad 7
Pruebas de Compatibilidad 8
Criterios de Entrada y Salida 8
Personal involucrado. 8
Planificación y Cronograma 9
Ejecución de pruebas y fichas. 10
Control de Cambios y Revisión 11
Riesgos 14

Tablas de contenido

Tabla 1. Tipos de prueba 5
Tabla 2. Pruebas funcionales 6
Tabla 3. Pruebas de rendimiento 7
Tabla 4. Pruebas de funcionalidad 7
Tabla 5. Pruebas de compatibilidad 8
Tabla 6. Información del Analista, diseño y programador 8
Tabla 7. Información del Analista, diseño y programador 9
Tabla 8. Actividades de pruebas 9
Tabla 9. Ficha de prueba de funcionalidad 10
Tabla 10. Ficha - Evaluación de Rendimiento 10
Tabla 11. Ficha - Evaluación de Usabilidad 11
Tabla 12. Control de cambios. 11
Tabla 13 Pruebas de Caja Negra 12
Tabla 14 Pruebas de Caja Blanca 13
Tabla 15. Riesgos 14

Introducción

Alcance

El sistema para generar Time Lapse a partir de imágenes extraídas del GOES-16 proporciona una plataforma de información en línea para la generación y visualización de animaciones de datos climáticos en tiempo real de la provincia de Bolívar, incluyendo:

- **Generación de animaciones**: Procesamiento de imágenes satelitales y superposición de capas para crear videos time lapse.
- **Subida automática:** Carga de los videos generados a un servidor web.
- **Visualización en línea:** Interfaz web que permite la visualización interactiva y ordenada de las animaciones generadas.

Referencias.

- Documento SRS
- Norma ISO/IEC/IEEE 29119

Objetivo

- Garantizar que la aplicación web "Visor Grintec" funciona de manera eficiente, precisa y segura, cumpliendo con los requisitos funcionales y no funcionales definidos, a través de la planificación y ejecución de un conjunto exhaustivo de pruebas

Tipos de Pruebas

Tabla 1. Tipos de prueba

Tipo de Prueba	Descripción	Herramientas/Recursos
Pruebas Funcionales	Verificación de las funcionalidades específicas según los requisitos.	Navegadores, herramientas de gestión de pruebas (Jira, TestRail)
Pruebas de Rendimiento	Evaluación de la carga y respuesta del sistema bajo diferentes condiciones.	JMeter, herramientas de análisis de rendimiento

Pruebas de Usabilidad	Análisis de la facilidad de uso y la experiencia del usuario final.	Evaluación de usuarios, encuestas, herramientas de usabilidad
Pruebas de Compatibilidad	Verificación de la funcionalidad en diferentes navegadores y dispositivos.	Navegadores, herramientas de pruebas cross-browser (BrowserStack)

Elaborado por: Bryan Gaibor y Franklin García.

Estrategia de Pruebas

Pruebas Funcionales

Tabla 2. Pruebas funcionales

Caso de Prueba	Descripción	Resultado Esperado	Estado
CP1 - Visualización del Visor	Verificar que el visor muestra el time lapse correctamente.	El visor debe mostrar el time lapse en tiempo real.	Realizado
CP2 - Menú de Selección de Canales	Probar la selección de diferentes canales en el menú.	El visor debe cambiar al canal seleccionado.	Realizado
CP3 - Controles de Reproducción	Validar los controles de reproducir, pausar, adelantar, retroceder, zoom y desplazamiento.	Cada control debe funcionar según lo esperado.	Realizado
CP4 - Historial de Videos	Comprobar la funcionalidad del historial de videos.	El historial debe mostrar videos guardados por fecha y canal.	Realizado

Pruebas de Rendimiento

Tabla 3. Pruebas de rendimiento

Prueba de Rendimiento	**Descripción**	**Criterio de Éxito**	**Estado**
PR1 - Tiempo de Carga Inicial	Medir el tiempo de carga del visor y la primera visualización.	El tiempo de carga debe ser inferior a 5 segundos.	Realizado
PR2 - Tiempo de Respuesta del Control	Evaluar el tiempo de respuesta de los controles de reproducción bajo condiciones normales.	Los controles deben responder en menos de Y segundos.	Realizado
PR3 - Pruebas de Estrés	Simular múltiples usuarios concurrentes para probar la estabilidad.	El sistema debe manejar 100 usuarios simultáneos sin fallos.	Realizado

Pruebas de Usabilidad

Tabla 4. Pruebas de funcionalidad

Prueba de Usabilidad	**Descripción**	**Criterio de Éxito**	**Estado**
PU1 - Navegación Intuitiva	Evaluar la facilidad de navegación a través del visor y menús.	Los usuarios deben poder navegar sin dificultad.	Realizado
PU2 - Comprensión de Controles	Probar si los usuarios comprenden fácilmente los controles de reproducción.	La mayoría de los usuarios deben comprender el uso de los controles.	Realizado

Pruebas de Compatibilidad

Tabla 5. Pruebas de compatibilidad

Prueba de Compatibilidad	**Descripción**	**Criterio de Éxito**	**Estado**
PC1 - Navegadores Principales	Verificar el funcionamiento en Chrome, Firefox, Edge, Safari.	La aplicación debe funcionar correctamente en todos los navegadores.	Realizado
PC2 - Dispositivos Móviles	Evaluar la funcionalidad en diferentes dispositivos móviles (Android, iOS).	La aplicación debe ser funcional en los principales dispositivos móviles.	Realizado
PC3 - Resoluciones de Pantalla	Probar la visualización en diferentes resoluciones de pantalla.	El visor debe adaptarse a diversas resoluciones sin problemas.	Realizado

Elaborado por: Bryan Gaibor y Franklin García.

Criterios de Entrada y Salida

Criterio de entrada

Desarrollo completo del visor y funcionalidades. Entorno de pruebas estable.

Criterio de salida

Todas las pruebas completadas con éxito. Defectos críticos resueltos. Cumplimiento de estándares de calidad.

Personal involucrado.

Tabla 6. Información del Analista, diseño y programador

Nombre	Bryan Gaibor
Rol	Analista, diseñador y programador
Responsabilidad	Recopilación de datos, diseño y desarrollo de la aplicación web “Visor GRINTEC”.
Información de contacto	brygaibor@mailes.ueb.edu.ec

Tabla 7. Información del Analista, diseño y programador

Nombre	Franklin Garcia
Rol	Analista, diseñador y programador
Responsabilidad	Recopilación de datos, diseño y desarrollo de la aplicación web “Visor GRINTEC”.
Información de contacto	fgarcia@mailes.ueb.edu.ec

Elaborado por: Bryan Gaibor y Franklin García

Planificación y Cronograma

1. Elaboración de la ficha de observación.
2. Realización de las pruebas.
3. Registro de los resultados obtenidos en las pruebas.

- **Inicio:** 17 de septiembre
- **Finalización:** 26 de septiembre

Actividades de Pruebas:

Tabla 8. Actividades de pruebas

Actividad	**Día 1**	**Día 2**	**Día 3**	**Día 4**	**Día 5**
Elaboración de la ficha de observación	X				
Realización de la prueba		X	X	X	X
Registro de los resultados de las pruebas	X	X	X	X	X

Elaborado por: Bryan Gaibor y Franklin García.

Ejecución de pruebas y fichas.

Ficha de prueba de funcionalidad

Tabla 9. Ficha de prueba de funcionalidad

Elemento	**Descripción**					
Objetivo	Probar la funcionalidad del producto					
Producto	"Visor Grintec"					
Fecha	17/09/2024					
Observador	Dr. Henry Vallejo					
Calificación	1 = no cumple, 5 = cumple satisfactoriamente					
Caso de prueba	1	2	3	4	5	Observaciones
Caso de prueba 01					**X**	La selección de canal y fecha funciona correctamente desde la interfaz.
Caso de prueba 02					**X**	Las animaciones se reproducen con fluidez, permitiendo pausar, adelantar y hacer zoom.

Elaborado por: Bryan Gaibor y Franklin García

Ficha - Evaluación de Rendimiento

Tabla 10. Ficha - Evaluación de Rendimiento

Elemento	Descripción					
Objetivo	Evaluar el rendimiento del producto					
Producto	"Visor Grintec"					
Fecha	17/09/2024					
Observador	Dr. Henry Vallejo					
Calificación	1 = muy lento, 5 = excelente					
Caso de Prueba	1	2	3	4	5	Observaciones
Caso de prueba 01: Tiempo de selección de canal y fecha					X	La selección de canal y fecha ocurre en un tiempo razonable, con respuesta eficiente.
Caso de prueba 02: Tiempo de reproducción de animaciones				X		La reproducción de animaciones es fluida, con ligeras demoras en la inicialización.

Ficha - Evaluación de Usabilidad

Tabla 11. Ficha - Evaluación de Usabilidad

Elemento	**Descripción**
Objetivo	Evaluar la usabilidad del producto
Producto	"Visor Grintec"
Fecha	19/09/2024
Observador	Dr. Henry Vallejo
Calificación	1 = muy deficiente, 5 = excelente

Caso de prueba	**1**	**2**	**3**	**4**	**5**	**Observaciones**
Caso de prueba 01: Visualización de Información					X	La información se visualiza correctamente en la página web, con buena claridad y accesibilidad.
Caso de prueba 02: Reproducción de Videos Generados					X	Los videos generados se reproducen sin interrupciones y los controles responden de forma efectiva.

Elaborado por: Bryan Gaibor y Franklin García

Control de Cambios y Revisión

Tabla 12. Control de cambios.

Aspecto	**Descripción**	**Proceso**
Gestión de Cambios	Proceso para manejar cambios en los requisitos de prueba.	Solicitudes de cambio, revisión y aprobación.
Revisión del Plan	Revisiones periódicas del plan de pruebas, ajustes necesarios.	Revisión semanal, actualizaciones del plan.

Elaborado por: Bryan Gaibor y Franklin García.

Pruebas de Caja Negra

Las pruebas de caja negra se centran en evaluar las funcionalidades del sistema basado en los requerimientos establecidos, sin considerar cómo está construido

internamente. Estas pruebas permiten verificar que el sistema cumple con los requisitos funcionales especificados en los casos de uso.

Se realizo pruebas exclusivamente para los siguientes requerimientos del sistema:

1. Visualización de información en tiempo real (RF01).
2. Reproducción fluida de videos generados y funcionamiento adecuado de los controles (RF02).

Tabla 13 Pruebas de Caja Negra

Caso de Prueba	**Descripción**	**Entradas**	**Resultado Esperado**	**Resultado Obtenido**	**Estado**
01: Visualización de Información	Verificar que la información en tiempo real se visualice de forma clara y actualizada en la página web.	Acceso a la página principal	La información debe ser visible, clara, y actualizarse automáticamente sin errores.	La información se visualizó claramente y se actualizó sin fallos.	**Aprobado**
02: Reproducción de Videos	Comprobar que los videos generados se reproduzcan sin interrupciones y los controles funcionen correctamente.	Seleccionar un video y usar los controles	Los videos deben reproducirse de manera fluida, con controles de reproducción operativos.	Reproducción fluida y controles funcionales.	**Aprobado**

Elaborado por: Bryan Gaibor y Franklin García

Pruebas de Caja Blanca

Las pruebas de caja blanca se centran en evaluar el funcionamiento interno del sistema, revisando la lógica del código, las estructuras y los flujos. En este caso, se analizarán los

scripts y procesos que soportan los requerimientos clave del sistema para garantizar su correcto funcionamiento.

Se realizarán pruebas exclusivamente para los siguientes requerimientos del sistema:

1. Visualización de información en tiempo real (flujos y actualización de datos).
2. Reproducción fluida de videos generados y funcionamiento de controles (lógica de reproducción y manejo de eventos).

Tabla 14 Pruebas de Caja Blanca

Caso de Prueba	**Descripción**	**Entradas**	**Proceso Interno Evaluado**	**Resultado Esperado**	**Resultado Obtenido**	**Estado**
01: Visualización de Información	Verificar que el script actualice correctamente la información en tiempo real en la base de datos.	Datos enviados por sensores remotos	Lógica de actualización de datos y su reflejo en la página web.	Los datos se actualizan en la base y se reflejan en tiempo real en la interfaz.	Los datos se actualizaron correctamente.	**Aprobado**
02: Reproducción de Videos	Comprobar que la lógica de reproducción de videos sea eficiente y los controles se procesen correctamente.	Reproducción de un video con controles (play, pause, etc.)	Manejo de eventos para controles y bucles de reproducción.	La reproducción es fluida, y los eventos se procesan sin errores ni retrasos.	Los eventos y la reproducción funcionaron como se esperaba.	**Aprobado**

Elaborado por: Bryan Gaibor y Franklin García

Riesgos

Tabla 15. Riesgos

Riesgo	Descripción	Impacto	Probabilidad	Plan de Contingencia
Falta de Recursos Humanos	Insuficiencia de personal para realizar todas las pruebas planificadas.	Alto: Puede retrasar el cronograma de pruebas.	Media	Reasignar tareas, ajustar el alcance de pruebas.
Problemas Técnicos en el Entorno de Pruebas	Fallos en el entorno de pruebas que afectan la ejecución de las pruebas.	Alto: Puede detener las pruebas hasta que se resuelvan.	Alta	Establecer un entorno de pruebas de respaldo, solucionar problemas técnicos rápidamente.
Cambios en los Requisitos	Modificaciones en los requisitos durante la fase de pruebas.	Medio: Puede requerir la re-ejecución de pruebas.	Media	Implementar un proceso de gestión de cambios, ajustar el plan de pruebas según sea necesario.
Errores Críticos en la Aplicación	Identificación de errores críticos que afectan la funcionalidad principal del visor.	Alto: Puede impedir el lanzamie	Media	Priorizar la resolución de errores críticos, coordinar con el equipo de

		nto del producto.		desarrollo para soluciones rápidas.
Fallas Pruebas Automati-zadas	Problemas con la ejecución de pruebas automatizadas debido a fallos en los scripts o herramientas.	Medio: Retraso en las pruebas automat izadas.	Baja	Revisar y actualizar los scripts de prueba, realizar pruebas manuales si es necesario.
Incompatibil idad con Navegadores Dispositivos	Problemas de compatibilidad en navegadores o dispositivos que no se detectaron previamente.	Medio: Afecta a la experien cia del usuario en ciertas platafor mas.	Media	Realizar pruebas de compatibilidad adicionales, ajustar la aplicación para resolver problemas detectados.
Rendimiento Deficiente Bajo Carga	El visor puede no funcionar adecuadamente bajo condiciones de alta carga de usuarios.	Alto: Puede afectar la disponibilidad del servicio.	Media	Optimizar el rendimiento de la aplicación, realizar ajustes basados en resultados de pruebas de carga.

Elaborado por: Bryan Gaibor y Franklin García.

7. MANUAL TÉCNICO

INGENIERÍA EN SOFTWARE

RSR-MET



VISOR GRINTEC

Manual Técnico

PROYECTO: TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR.

Versión 1.0

ELABORADO POR:

Bryan Hernán Gaibor Naranjo

Franklin Sebastián García Villares

CONTENIDO

1 Propósito 4

2 Requerimientos de desarrollo 4

2.1 Requerimientos de hardware 4

2.2 Requerimientos de software 4

3 Herramientas utilizadas para el desarrollo 4

4 Estructura del Proyecto 5

4.1 Componentes principales 5

4.2 Estructura de archivos y carpetas 5

5 Descripción técnica del funcionamiento 6

5.1 Procesamiento de imágenes satelitales 6

5.2 Visualización de los videos en la web 6

6 Instrucciones para la ejecución 6

6.1 Configuración inicial 6

6.2 Ejecución de scripts 7

7 Configuración del Crontab 7

8 Mantenimiento y actualización 7

8.1 Logs y errores comunes 7

8.2 Futuras ampliaciones 7

9 Configurar Express en el puerto 443 8

9.1 Obtener Certificados SSL 8

9.2 Modificar el código de Express para HTTPS 8

10.1 Redirigir el puerto 443 a uno mayor 9

10.2 Cambiar permisos para PM2 9

10.3 Guardar la configuración de PM2 9

1 Propósito

Este manual técnico tiene como finalidad proporcionar información detallada sobre la estructura y funcionamiento del sistema desarrollado para la generación de animaciones time-lapse a partir de imágenes satelitales del satélite GOES-16. La solución incluye el procesamiento automático de imágenes, la generación de videos y su publicación para visualización en línea.

2 Requerimientos de desarrollo

2.1 Requerimientos de hardware

- Raspberry Pi 5
- Memoria RAM: 4GB o más
- Almacenamiento: 32GB o superior
- Conexión a Internet estable para la carga de videos a servidores FTP

2.2 Requerimientos de software

- Sistema operativo: Debian 12 (para el servidor web)
- Python 3.10 o superior
- Bibliotecas requeridas (ver archivo requirements.txt)
- Apache2 con proxy inverso configurado
- PM2 para mantener activo el servidor Express

3 Herramientas utilizadas para el desarrollo

- **Lenguaje de programación:** Python, JavaScript (Node.js con Express.js)
- **Frameworks y bibliotecas:**
    - OpenCV (procesamiento de imágenes)
    - MoviePy (generación de videos)
    - PIL (manejo de imágenes)
- **Servidor Web:** Apache2 con proxy inverso
- **Control de procesos:** PM2

- **Sistema de gestión de versiones:** Git

4 Estructura del Proyecto

4.1 Componentes principales

- **Recorte.py:** Se encarga del procesamiento de las imágenes satelitales.
- **Enviar_ftp.py:** Automáticamente sube los videos generados a un servidor FTP.
- **Visor Grintec (Node.js + Express.js):** Interfaz web que permite la visualización de los videos generados.

4.2 Estructura de archivos y carpetas

RasberryPI

```
├── recorte.py
├── enviar_ftp.py
├── config.cfg
├── logs/
  ├── recorte.log
  └── enviar.log
```

Servidor Ftp

```
├── ftp/
    └── enviar.log
```

ServidorWeb

```
├──VisorGrintec/
  ├── public/
      ├── fonts/
      ├── imgs/
      ├── vendor/
      ├──index.html
      ├──script.js
```

```
        ├──styles.css
└── app.js
```

5 Descripción técnica del funcionamiento

5.1 Procesamiento de imágenes satelitales

El script recorte.py se ejecuta periódicamente a través de un crontab. Este script:

- Recorta las imágenes satelitales y aplica un modelo de superresolución (FSRCNN).
- Superpone una imagen de las fronteras de las provincias del Ecuador.
- Genera un video time-lapse a partir de las imágenes recortadas y procesadas.

Subida automática de videos al servidor FTP

El script enviar_ftp.py se encarga de:

- Leer el archivo video_to_upload.txt donde están listados los videos generados.
- Conectar con el servidor FTP configurado en config.cfg.
- Subir automáticamente los videos generados al servidor FTP.
- Eliminar los archivos locales una vez que la subida ha sido exitosa.

5.2 Visualización de los videos en la web

El Visor Grintec, desarrollado con Node.js y Express.js, permite:

- Navegar entre las diferentes bandas del satélite GOES-16.
- Visualizar los videos time-lapse generados por fechas y horas específicas.

6 Instrucciones para la ejecución

6.1 Configuración inicial

1. Configurar los valores en config.cfg, incluyendo rutas de imágenes y credenciales del servidor FTP, todo detallado dentro del archivo cfg.
2. Instalar las dependencias necesarias dentro del entorno virtual Python creado ejecutando pip install -r requirements.txt.

6.2 Ejecución de scripts

- recorte.py se ejecuta automáticamente cada 30 minutos.
- enviar_ftp.py se ejecuta 10 minutos después de recorte.py para subir los videos al servidor.

7 Configuración del Crontab

El sistema utiliza crontab para automatizar la ejecución de los scripts. A continuación, se detalla la configuración del crontab utilizada.

Pegar en archivo cron lo siguiente:

Ejecuta recorte.py cada 30 minutos y guarda la salida en un archivo de log

*/30 * * * * /Ruta de archivo pyhton/recorte.py >> /Ruta de /logs/recorte.log 2>&1

Ejecuta enviar_ftp.py 10 minutos después de recorte.py

10,40 * * * * /Ruta de entorno virtual Python/venv/bin/python / Ruta de archivo pyhton//enviar_ftp.py >> //Ruta de /logs /enviar.log 2>&1

- recorte.py se ejecuta cada 30 minutos para procesar las imágenes.
- enviar_ftp.py se ejecuta 10 minutos después de recorte.py para garantizar que los videos ya hayan sido generados.

También se utiliza un cron para el servidor donde van alojados los videos, cada video dura un máximo de 8 dias para que no se sobrecargue el almacenamiento del servidor.
0 2 * * * find /ruta/a/directorio/de/videos -type f -mtime +8 -exec rm {} \;

8 Mantenimiento y actualización

8.1 Logs y errores comunes

- Revisar los archivos de log en la carpeta logs/ para diagnosticar errores.
- Errores comunes incluyen fallos en la conexión FTP o en el procesamiento de imágenes.

8.2 Futuras ampliaciones

Se puede agregar la posibilidad de generar videos con resoluciones más altas o agregar nuevas bandas de imágenes satelitales según sea necesario.

9 Configurar Express en el puerto 443

El puerto 443 es el estándar para conexiones HTTPS. Si vas a usarlo, necesitarás un certificado SSL. Sigue estos pasos:

9.1 Obtener Certificados SSL

- Puedes usar servicios como Let's Encrypt para obtener certificados SSL gratuitos.
- Una vez generados, obtendrás dos archivos:
  - **Certificado público (fullchain.pem)**
  - **Llave privada (privkey.pem)**

9.2 Modificar el código de Express para HTTPS

Agrega soporte para HTTPS en tu aplicación:

```
const app = express();

// Leer certificados SSL
const sslOptions = {
  key: fs.readFileSync('/ruta/a/privkey.pem'),
  cert: fs.readFileSync('/ruta/a/fullchain.pem'),
};

// Configura HTTPS en el puerto 443
https.createServer(sslOptions, app).listen(443, () => {
  console.log('Servidor HTTPS corriendo en el puerto 443');
});
```

Asegúrate de que tu dominio apunte al servidor

Tu dominio debe estar configurado correctamente para apuntar a la dirección IP del servidor. Esto se hace desde el panel de DNS del proveedor de tu dominio.

10 Configurar PM2 para ejecutar en el puerto 443 sin usar root

Por defecto, solo el **root** puede usar puertos menores a 1024 (como el 443). Para evitar esto:

10.1 Redirigir el puerto 443 a uno mayor

Usa **iptables** para redirigir el tráfico del puerto 443 al puerto donde corre tu aplicación, por ejemplo, el **8443**:

sudo iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 8443

Este comando redirige automáticamente las solicitudes HTTPS al puerto 8443, que puede ser utilizado por usuarios no root.

10.2 Cambiar permisos para PM2

PM2 puede ser configurado para que no necesite permisos de root:

1. **Ejecuta PM2 con un usuario no root**
    - Si PM2 fue configurado como **root**, detén la aplicación:

pm2 stop all

pm2 unstartup

    - Cambia al usuario no root y configura PM2:

sudo su - <tu-usuario-no-root>

pm2 startup

2. **Inicia la aplicación como usuario no root**

pm2 start app.js --name mi-app

10.3 Guardar la configuración de PM2

Asegúrate de que PM2 recuerde tu configuración después de un reinicio:

pm2 save

Y habilita el inicio automático:

pm2 startup

8. FICHAS DE OBSERVACIÓN

Fichas de Observación del Proyecto

Ficha de Observación: Funcionamiento General

Fecha: 14/09/2024

Área Evaluada: Funcionamiento de la aplicación web

Descripción: Verificar la interacción entre los componentes del sistema (cliente-servidor) y la correcta generación de animaciones.

Aspectos Observados:

1. Las animaciones se generan y visualizan correctamente.
2. El tiempo de carga de las animaciones es aceptable para el usuario final.
3. La interfaz web responde a las solicitudes del usuario sin interrupciones.

Incidencias:

1. No se muestran errores en la generación o visualización de las animaciones.
2. El tiempo de respuesta fue ligeramente superior al esperado en algunas ocasiones, pero dentro de límites tolerables para el usuario.

Ficha de Observación: Scripts en Raspberry Pi

Fecha: 14/09/2024

Área Evaluada: Scripts recorte.py y enviar_ftp.py

Descripción: Monitorear el procesamiento y envío de archivos desde el cliente al servidor.

Aspectos Observados:

1. recorte.py ejecuta el procesamiento de imágenes correctamente.
2. enviar_ftp.py transfiere los archivos al servidor sin pérdida de datos.
3. Los logs generados son claros y contienen información relevante.

Incidencias:

1. No se detectaron archivos que no fueron procesados.
2. No hubo problemas en la conexión FTP durante las transferencias.

Ficha de Observación: Servidor Express

Fecha: 14/09/2024

Área Evaluada: Configuración y desempeño del servidor Node.js

Descripción: Evaluar la respuesta del servidor Express a las solicitudes del cliente.

Aspectos Observados:

1. Las rutas configuradas responden correctamente.
2. Las solicitudes HTTP/HTTPS son manejadas de manera eficiente.
3. El rendimiento del servidor bajo carga es aceptable.

Incidencias:

1. No se presentarán caídas ni errores durante las pruebas.
2. El servidor no saturó ni la CPU ni la memoria durante el proceso de carga.

Ficha de Observación: Interfaz del Usuario

Fecha: 14/09/2024

Área Evaluada: Sitio web grintec.org/visor

Descripción: Observar la usabilidad y experiencia del usuario.

Aspectos Observados:

1. La interfaz es intuitiva y fácil de usar.
2. Los usuarios pueden seleccionar y visualizar animaciones sin dificultades.
3. Los elementos gráficos (botones, selectores, etc.) funcionan como se espera.

Incidencias:

1. No se detectaron componentes de la interfaz que no funcionarán.
2. No hubo problemas de compatibilidad con los navegadores probados.

Ficha de Observación: Logs del Cron

Fecha: 14/09/2024

Área Evaluada: Tareas programadas

Descripción: Monitorear el correcto funcionamiento de las tareas programadas en cron.

Aspectos Observados:

1. Los scripts se ejecutan en los tiempos programados (12 am, 6 am, 12 pm, 6 pm).
2. Los logs reflejan el estado de las tareas sin errores.
3. La transferencia de archivos ocurre dentro de los tiempos esperados.

Incidencias:

1. En la primera prueba, la tarea programada no se ejecutó debido a que cron no tenía permisos para escribir en el servidor. Este problema fue solucionado al ajustar los permisos adecuados.
2. No se registraron errores adicionales en los registros de cron.

Ficha de Observación: Rendimiento

Fecha: 14/09/2024

Área Evaluada: Desempeño general del sistema

Descripción: Evaluar el rendimiento bajo condiciones de uso normal y carga elevada.

Aspectos Observados:

1. El servidor responde adecuadamente bajo carga de múltiples usuarios.
2. Los tiempos de procesamiento de imágenes son óptimos.
3. El sistema escala adecuadamente a medida que aumenta el volumen de datos.

Incidencias:

1. No se registraron tiempos de espera largos durante las pruebas.
2. El sistema no colapsó bajo alta demanda y mantuvo un rendimiento estable.

9. CERTIFICADO DE ENTREGA DEL PRODUCTO

DR. HENRY FERNANDO VALLEJO BALLESTEROS, COORDINADOR RESPONSABLE DEL GRUPO INVESTIGACION TECNOLOGÍAS APLICADAS.

A petición de la parte interesada,

CERTIFICA

Que los señores **Gaibor Naranjo Bryan Hernán y García Villares Franklin Sebastián**, portadores de la cédula de identidad **1752326007 y 0202077640** respectivamente, entregaron la aplicación web correspondiente al proyecto de su trabajo de integración curricular titulado, **"TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR "**, cumpliendo con las especificaciones solicitadas.

Es cuanto puedo certificar, y se extiende la presente al interesado para el uso que estime conveniente, salvo asuntos judiciales.

Guaranda, 09 de diciembre del 2024

DR. HENRY VALLEJO B.

Coordinador responsable GTA

10. CERTIFICADO DE ANÁLISIS DE COINCIDENCIA NO ACCIDENTAL

DR. HENRY FERNANDO VALLEJO BALLESTEROS EN CALIDAD DE DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR,

CERTIFICA

Que el trabajo de integración curricular denominado ""**TIME LAPSE A PARTIR DE IMÁGENES SATELITALES EN TIEMPO REAL DEL GOES-16, EN LA PROVINCIA BOLÍVAR** "," presentado por **Gaibor Naranjo Bryan Hernán y García Villares Franklin Sebastián** estudiantes de la **carrera de Software** pasó el análisis de coincidencia no accidental en la herramienta TURNITIN, reflejando un **porcentaje de similitud del 3%**, como se puede evidenciar en el documento adjunto.

Guaranda, 06 de diciembre del 2024

Atentamente,

Dr. Henry Vallejo B.
Director

BRYAN GAIBOR

TESIS TIME LAPSE GOES 16.docx

My Files

My Files

Universidad Estatal de Bolivar

Detalles del documento

Identificador de la entrega

trn:oid:::3117:413178051

Fecha de entrega

5 dic 2024, 10:38 p.m. GMT-5

Fecha de descarga

5 dic 2024, 11:38 p.m. GMT-5

Nombre de archivo

TESIS TIME LAPSE GOES 16.docx

Tamaño de archivo

1.9 MB

139 Páginas

21,435 Palabras

124,689 Caracteres

3% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Trabajos entregados

Exclusiones

- N.º de fuentes excluidas
- N.º de coincidencias excluidas

Fuentes principales

3% Fuentes de Internet

0% Publicaciones

0% Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alertas de integridad para revisión

No se han detectado manipulaciones de texto sospechosas.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

Fuentes principales

3% Fuentes de Internet
0% Publicaciones
0% Trabajos entregados (trabajos del estudiante)

Fuentes principales

Las fuentes con el mayor número de coincidencias dentro de la entrega. Las fuentes superpuestas no se mostrarán.

1 Internet
eos.com 0%

2 Internet
www.dominiodelasciencias.com 0%

3 Internet
www.climatol.eu 0%

4 Internet
ribuni.uni.edu.ni 0%

5 Internet
www.buenastareas.com 0%

6 Internet
patrimonio.uo.edu.cu 0%

7 Internet
repositorio.usm.cl 0%

8 Internet
eprints.uanl.mx 0%

9 Internet
docs.aws.amazon.com 0%

10 Internet
online-tesis.com 0%

11 Internet
mundo-hackers.weebly.com 0%

12 Internet

docplayer.net 0%

13 Internet

sbgrid.org 0%

14 Publicación

Beltran, Susana María. "Investigating Mechanotransduction Behaviors of Natural... 0%

15 Internet

orcid.org 0%

16 Internet

pt.scribd.com 0%

17 Internet

www.prnewswire.com 0%

18 Internet

books.google.com.co 0%

19 Internet

ftp.gwdg.de 0%

20 Internet

indotel.org.do 0%

21 Internet

riunet.upv.es 0%

22 Internet

www.genwords.com 0%

23 Publicación

Sanchez Peña, Diana Carolina. "Sistema de Información para la Gestión Integral d... 0%

24 Internet

ed.team 0%

25 Internet

qdoc.tips 0%

26 Internet
repositorio.uam.es 0%

27 Internet
repository.usta.edu.co 0%

28 Internet
ridum.umanizales.edu.co 0%

29 Internet
www.tecni.com 0%

30 Internet
www.xmlstarterkit.com 0%

31 Internet
transportesynegocios.wordpress.com 0%

32 Internet
ignaciosantiago.com 0%

33 Internet
zaguan.unizar.es 0%

34 Internet
dspace.unach.edu.ec 0%

35 Internet
gredos.usal.es 0%

36 Internet
renati.sunedu.gob.pe 0%

37 Internet
www.eu.microsoft.com 0%