

**FACULTAD DE INGENIERÍA**

Escuela Académico Profesional de Ingeniería Mecatrónica

Tesis

**Diseño de un aplicativo para la interpretación del panel  
naranja, implementado en un "Vant" para el nivel  
advertencia Del Cuerpo General de Bomberos  
de la región Junín**

Harvith Brayan Echevarría Zenayuca  
Lenin Kepler Guerra Galarza

Para optar el Título Profesional de  
Ingeniero Mecatrónico

Huancayo, 2022

Repositorio Institucional Continental  
Tesis digital



Esta obra está bajo una Licencia "Creative Commons Atribución 4.0 Internacional" .

## **ASESOR**

Ma. Ing. Rafael De La Cruz Casaño

## **AGRADECIMIENTOS**

En primer lugar, dar las gracias a Dios y a nuestros padres quienes nos brindaron la posibilidad de estudiar una carrera universitaria y también estuvieron presentes a lo largo de nuestra vida académica y gracias a ellos hemos logrado concluir nuestros estudios superiores.

De igual manera, dar las gracias a nuestros maestros de la Universidad Continental por los conocimientos y valores compartidos que asistieron a nuestro crecimiento profesional y personal.

Además, dar las gracias a nuestro asesor de tesis quien nos orientó y brindó sus conocimientos para resolver nuestras dudas, gracias a ello pudimos concluir de manera satisfactoria esta investigación.

Por último, dar las gracias a todos nuestros colegas y familiares que estuvieron asistiéndonos a lo largo de nuestra vida universitaria.

## **DEDICATORIA**

A nuestros padres, quienes nos proporcionaron su respaldo incondicional, gracias a ellos hemos logrado concluir de manera satisfactoria esta investigación.

# ÍNDICE GENERAL

<b>Asesor</b> .....	<b>ii</b>
<b>Agradecimientos</b> .....	<b>iii</b>
<b>Dedicatoria</b> .....	<b>iv</b>
<b>Índice general</b> .....	<b>v</b>
<b>Índice de figuras</b> .....	<b>viii</b>
<b>Índice de tablas</b> .....	<b>x</b>
<b>Resumen</b> .....	<b>xi</b>
<b>Abstract</b> .....	<b>xii</b>
<b>Introducción</b> .....	<b>xiii</b>
<b>CAPÍTULO I</b> .....	<b>16</b>
<b>PLANTEAMIENTO DEL ESTUDIO</b> .....	<b>16</b>
1.1 Planteamiento y formulación del problema .....	16
1.1.1 Problema general.....	18
1.1.2 Problemas específicos .....	18
1.2 Objetivos.....	19
1.2.1 Objetivo general .....	19
1.2.2 Objetivos específicos .....	19
1.3 Justificación .....	19
1.3.1 Justificación social .....	19
1.3.2 Justificación teórica.....	19
1.3.3 Justificación práctica .....	19
1.3.4 Justificación metodológica .....	20
<b>CAPÍTULO II</b> .....	<b>21</b>
<b>MARCO TEÓRICO</b> .....	<b>21</b>
2.1 Antecedentes del problema .....	21
2.1.1 Internacionales .....	21
2.1.2 Nacionales .....	24
2.2 Bases teóricas .....	27
2.2.1 Visión artificial .....	27
2.2.2 Software de procesamiento de imágenes.....	27
2.2.3 OpenCV .....	27

2.2.4 Python .....	27
2.2.4.1 Variables en Python.....	28
2.2.4.2 Librerías en Python.....	28
2.2.4.3 Librería Tkinter .....	28
2.2.4.4 Librería Pytesseract.....	28
2.2.4.5 Funciones en Python .....	28
2.2.4.6 Sentencias condicionales .....	29
2.2.5 Estudio de tecnologías .....	30
2.2.6 Aplicaciones Web.....	31
2.2.7 HTML5 .....	31
2.2.8 CSS3.....	32
2.2.9 Java Script .....	32
2.2.10 Aplicaciones híbridas .....	33
2.2.11 Diagrama de flujo .....	33
2.3 Definición de términos básicos .....	34
<b>CAPÍTULO III.....</b>	<b>35</b>
<b>METODOLOGÍA.....</b>	<b>35</b>
3.1 Tipo de investigación .....	35
3.2 Metodológica aplicada para el desarrollo de la solución .....	35
<b>CAPÍTULO IV .....</b>	<b>42</b>
<b>ANÁLISIS Y DISEÑO DE LA SOLUCIÓN .....</b>	<b>42</b>
4.1. Identificación de los requerimientos .....	42
4.1.1 Detallar el problema .....	42
4.1.2 Resumen analítico de la investigación.....	43
4.1.3 Ensayo crítico del tema de investigación .....	49
4.1.4 Lista de exigencias.....	50
4.2 Análisis de la solución.....	50
4.2.1 Black-Box .....	50
4.2.2 Caja blanca .....	51
4.2.3 Matriz morfológica .....	53
4.2.4 Evaluación técnico – económica del concepto de solución.....	58
4.2.4.1 Valoración técnica .....	58
4.2.4.2 Valoración económica .....	58
4.2.4.3 Selección de sistema óptimo .....	59

4.3 Etapa de diseño.....	60
4.3.1 Programación de código base .....	60
4.3.1.1 Algoritmo de funcionamiento .....	60
4.3.1.2 Código de funcionamiento .....	61
4.3.2 Programación de código web .....	74
4.3.2.1 Código de página principal .....	74
4.3.2.2 Código de la página del panel de información .....	86
<b>CAPÍTULO V .....</b>	<b>91</b>
<b>CONSTRUCCIÓN.....</b>	<b>91</b>
5.1 Construcción.....	91
5.1.1 Inicio del aplicativo .....	91
5.1.2 Identificador OCR.....	93
5.1.3 Búsqueda manual .....	94
5.1.4 Almacenamiento de datos .....	96
5.1.5 Pagina web .....	96
5.2 Pruebas y resultados .....	97
<b>Conclusiones .....</b>	<b>105</b>
<b>Trabajos futuros.....</b>	<b>106</b>
<b>Lista de referencias .....</b>	<b>107</b>
<b>Anexos.....</b>	<b>110</b>

## ÍNDICE DE FIGURAS

Figura 1. Deflagración de cisterna.....	17
Figura 2. Ortomosaicos RGB construido con Pix4D volando a 120 metros de altura con 4 puntos de control foto insertados en los extremos de las flechas azules.....	22
Figura 3. Interfaz de la aplicación.....	23
Figura 4. Diseño final del robot minero .....	24
Figura 5. Diseño del VANT.....	24
Figura 6. Sistema integrado y operativo .....	25
Figura 7. Presentación final del sistema robótico .....	26
Figura 8. Diagrama de flujo condicional IF .....	29
Figura 9. Diagrama de flujo condicional IF – ELSE .....	29
Figura 10. Diagrama de flujo condicional IF – ELIF – ELSE .....	30
Figura 11 . Logotipo del estándar HTML 5. ....	32
Figura 12. Logotipo del estándar CSS 3.....	32
Figura 13. Logotipo de Java Script.....	33
Figura 14. Simbología de diagrama de flujo .....	34
Figura 15. Esquema de la adaptación de la metodología VDI 2206 .....	36
Figura 16. Caja negra. ....	38
Figura 17. Caja blanca. ....	38
Figura 18. Matriz morfológica.....	39
Figura 19. Caja blanca .....	52
Figura 20. Concepto de solución 1 .....	55
Figura 21. Concepto de solución 2.....	56
Figura 22. Concepto de solución 3.....	57
Figura 23. Gráfica de evaluación técnico – económica .....	59
Figura 24. Algoritmo de funcionamiento del aplicativo.....	60
Figura 25. Inicio de sesión del aplicativo .....	91
Figura 26. Ventana de error de datos.....	92
Figura 27. Menú de funciones del aplicativo .....	92
Figura 28. Lector inteligente.....	93
Figura 29. Número ONU identificado .....	93
Figura 30. Aplicativo online .....	94
Figura 31. Guía de emergencia online .....	94
Figura 32. Función de búsqueda manual .....	95
Figura 33. Resultados de búsqueda manual .....	95
Figura 34. Función de almacenamiento de datos .....	96
Figura 35. Página web del aplicativo .....	96
Figura 36. Guías de emergencia en la página web .....	97
Figura 37. Fuga de un contenedor .....	97
Figura 38. Inicio de sesión del aplicativo .....	98
Figura 39. Identificador OCR.....	98
Figura 40. Panel Naranja capturado.....	99
Figura 41. Datos identificados del Panel Naranja .....	99
Figura 42. Guía de emergencia asignada al “Mat-Pel” .....	100

Figura 43. Comunicación de datos al personal de bomberos .....	100
Figura 44. Intervención al área del siniestro .....	101
Figura 45. Fuga de “Mat-Pel” transportado en un vehículo .....	101
Figura 46. Inicio de sesión prueba 2 .....	102
Figura 47. Despegue del dron .....	102
Figura 48. Posicionamiento del dron .....	103
Figura 49. Datos extraídos del aplicativo.....	103
Figura 50. Intervención del personal de bomberos con los datos obtenidos del aplicativo .....	104
Figura 51. Programa utilizado para el desarrollo del aplicativo Visual Studio Code .....	111
Figura 52. Guía GRE2020.....	111
Figura 53. Número ONU 1075 (GLP) utilizado en las pruebas .....	112
Figura 54. Nivel advertencia brindando información obtenida del Aplicativo hacia el nivel Operaciones .....	112
Figura 55. Nivel Operaciones interviniendo una fuga de un “Mat-Pel” con los datos brindados del nivel Advertencia .....	112
Figura 56. Piloto especializado del dron .....	113
Figura 57. Evidencias del día de pruebas con apoyo de la compañía de bomberos César A. Habich Sosa N.º 30 .....	113

## ÍNDICE DE TABLAS

Tabla 1. Formato tabla de resumen analítico de la investigación .....	37
Tabla 2. Formato ensayo crítico del tema de investigación .....	37
Tabla 3. Formato de lista de exigencias .....	38
Tabla 4. Formato de evaluación técnica.....	39
Tabla 5. Formato de evaluación económica.....	40
Tabla 6. Análisis de imágenes multiespectrales adquiridas con vehículos aéreos no tripulados.....	43
Tabla 7. Evaluación de un prototipo de RPAS para el levantamiento topográfico con imágenes RGB .....	45
Tabla 8. Vehículo aéreo no tripulado para vigilancia y monitorización de incendios .....	47
Tabla 9. Tema de investigación .....	49
Tabla 10. Lista de exigencias.....	50
Tabla 11. Matriz morfológica .....	53
Tabla 12. Valoración técnica .....	58
Tabla 13. Valoración económica .....	58

## RESUMEN

La presente tesis tiene como objetivo general el diseño de un aplicativo para la interpretación del panel naranja, implementado en un "VANT" para el Nivel Advertencia del Cuerpo General de Bomberos de la región Junín, el cual tendrá la función de optimizar el tiempo de reconocimiento de un "Mat-Pel" desde una distancia segura salvaguardando la integridad del Cuerpo General de Bomberos en el cumplimiento de su deber.

La metodología aplicada fue una adaptación de la metodología VDI 2206, mediante la cual se desarrolló la programación del aplicativo, mediante 4 fases, las cuales fueron fase de especificaciones, funcional, de diseño y de codificación.

El desarrollo del aplicativo fue utilizando como fuente de alimentación un cargador de laptop por la portabilidad del sistema, para la captura del video fue mediante la cámara del vehículo aéreo no tripulado, utilizando como medio de transmisión el mando del Drone, recepcionando el video en vivo en una laptop donde se encontraba el aplicativo, el reconocimiento fue mediante el reconocimiento óptico de caracteres, el *Back-End* y el *Front-End* fue programado en Python, la API del aplicativo online fue desarrollada en HTML, CSS y JS utilizando una base de datos local mediante un archivo de texto, todo controlado mediante Windows 10.

El aplicativo permitió extraer el texto a través de una imagen capturada del video en vivo emitido de la cámara del dron, para su posterior identificación mediante la comparación del texto extraído con la base de datos, del cual se obtuvo los siguiente datos: el nombre del "Mat-Pel", la guía de emergencia asignada donde se encuentran los peligros potenciales, la seguridad pública y la respuesta ante emergencia de una fuga o derrame de un "Mat-Pel", consejos de intervención y la distancia de aislamiento para una evacuación rápida y segura.

**Palabras claves:** aplicativo, OCR, panel naranja, VANT, VDI2206

## ABSTRACT

The general objective of this thesis is the design of an application for the interpretation of the Orange Panel, implemented in a "VANT" for the Warning Level of the General Fire Department of the Junin region, which will have the function of optimizing the recognition time of a "Mat-Pel" from a safe distance, safeguarding the integrity of the General Fire Department in the fulfillment of their duty.

The methodology applied was an adaptation of the VDI 2206 methodology, through which we developed the programming of the application, through 4 phases, which were the specification phase, the functional phase, the design phase and the coding phase.

The development of the application was using as a power source a laptop charger for the portability of the system, for the video capture was through the camera of the UAV using as a transmission medium the control of the Drone, receiving the live video on a laptop where the application was located, the recognition was through Optical Character Recognition, the Back-End and Front-End was programmed in Python, the API of the online application was developed in HTML, CSS and JS using a local database through a text file all controlled by Windows10.

The application allowed us to extract the text through an image captured from the live video emitted from the drone camera, for subsequent identification by comparing the extracted text with the database, from which we obtained the following data: the name of the "Mat-Pel", the emergency guide assigned where the potential hazards are located, public safety and emergency response to a leak or spill of a "Mat-Pel", intervention tips and isolation distance for a quick and safe evacuation.

**Keywords:** application, OCR, Orange Panel, VANT, VDI2206

## INTRODUCCIÓN

La inseguridad a la que se expone el cuerpo general de bomberos al intervenir una fuga o derrame de materiales peligrosos “Mat-Pel” cuando realizan sus labores diarias amerita plantear soluciones tecnológicas para poder evitar accidentes en los cuales ellos tengan que exponer su integridad personal por el cumplimiento de su oficio.

El primer paso para controlar un siniestro “Mat-Pel” empieza en el nivel Advertencia del Cuerpo de Bomberos que tienen la función de identificar el “Mat-Pel” para transmitir los datos al nivel Operaciones que son los encargados de evaluar la situación para una posterior intervención.

Para poder realizar el proceso de identificación del “Mat-Pel” según el curso Primap del USAID se puede realizar de 2 formas mediante el panel de seguridad (Panel Naranja) o mediante los Documentos de transporte o embargo (MSDS) donde se encuentra el número ONU y mediante la guía GRE (Guía de Respuesta en caso de Emergencia) tomar las decisiones asertivas en el menor tiempo posible para controlar el siniestro y evitar accidentes de mayor envergadura. Pero la identificación mediante los documentos de transporte no es una opción permitida en una fuga o derrame de un “Mat-Pel”, ya que se encuentra en la cabina de conducción del vehículo de transporte y el nivel Advertencia tiene prohibido acercarse a la zona del siniestro por medidas de seguridad, entonces la forma más segura de identificación es mediante el Panel Naranja que se encuentra ubicada en diversas posiciones del vehículo donde además se encuentra el número ONU. Sin embargo, ocurren situaciones en las que no pueden observar con claridad el Panel Naranja ya que el nivel Advertencia se ubica en una posición segura fuera del área de fuga o derrame por el riesgo de exposición ante la fuga del “Mat-Pel”, entonces utilizan binoculares para poder visualizar el Panel Naranja, lo cual les disminuye el tiempo de respuesta ante el siniestro.

De esta forma, se justifica la investigación de acuerdo a la problemática planteada de inseguridad a la que se encuentran expuestos el personal de bomberos en el proceso de identificación del “Mat-Pel” que incita al desarrollo de una solución tecnológica para evitar exponer la integridad de los mismos al realizar estas tareas en ese entorno de alto riesgo, cumpliendo sus actividades del día a día.

Como objetivos, se plantea realizar el diseño de un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín, programar la interpretación del Panel Naranja y validar el funcionamiento del aplicativo.

Esta investigación utilizó el lenguaje de programación Python para programar el sistema de reconocimiento a partir de las imágenes obtenidas de la cámara del VANT mediante OCR, utilizando las librerías CV2, Pytesseract y Tkinter que permitió la identificación del Panel Naranja, donde se extrajeron los datos esenciales como el nombre del “Mat-Pel”, el área de aislamiento, consejos de intervención y la guía de emergencia asignada para una rápida reacción del nivel Advertencia del Cuerpo General de Bomberos de la región Junín.

En el capítulo I se desarrolló el planteamiento del estudio a partir de la problemática planteada iniciando con la formulación del problema, identificando los objetivos y justificando la investigación.

En el capítulo II se plantearon los antecedentes del problema, tanto internacionales como nacionales, además de las bases teóricas que sirvieron para el desarrollo del aplicativo y la definición de términos básicos utilizados en la investigación.

En el capítulo III se planteó la metodología a usar en el trabajo de investigación, que fue una adaptación de la VDI 2206 dividida en 4 fases.

En el capítulo IV se desarrolló la aplicación, utilizando la adaptación de la metodología VDI 2206, donde se inició con la identificación de los requerimientos, luego el análisis de la solución y la etapa de diseño del aplicativo.

En el capítulo V se desarrolló la construcción del aplicativo uniendo todo el apartado de diseño y programación para finalmente probar en una escenificación con el apoyo de la compañía de bomberos Cesar A. Habich Sosa N.º 30, obteniendo así, los resultados de la investigación.

Finalmente, se encuentran las conclusiones, trabajos futuros, lista de referencias y anexos.

# **CAPÍTULO I**

## **PLANTEAMIENTO DEL ESTUDIO**

### **1.1 Planteamiento y formulación del problema**

La inseguridad a la que se expone el cuerpo general de bomberos al intervenir una fuga o derrame de materiales peligrosos “Mat-Pel” cuando realizan sus labores diarias amerita plantear soluciones tecnológicas para poder evitar accidentes en los cuales ellos tengan que exponer su integridad personal por el cumplimiento de su oficio.

Se consideran materiales peligrosos a los sólidos, líquidos o gases que tienen la propiedad de provocar daños a personas, bienes o al ambiente (1).

Como en el suceso que ocurrió el 23 de enero del 2020 donde hubo 2 muertos y más de 40 heridos después de la deflagración de un camión cisterna, como se aprecia en la figura 1, que llevaba gas licuado de petróleo (GLP) que es considerado como un Mat-Pel de mucho riesgo (2).



**Figura 1. Deflagración de cisterna. Tomada del diario El Comercio**

En estas condiciones el personal de bomberos se encuentra expuesto al tratar de intervenir y controlar la fuga de ese GLP, tratando de encontrar el punto de escape que en cualquier momento podría iniciar la deflagración.

El primer paso para controlar un siniestro “Mat-Pel” empieza en el nivel Advertencia del cuerpo de bomberos que tienen la función de identificar el “Mat-Pel” para transmitir los datos al nivel Operaciones que son los encargados de intervenir.

Para poder realizar el proceso de identificación del “Mat-Pel” según el curso Primap del Usaid, se puede realizar de 2 formas, mediante el panel de seguridad (Panel Naranja) o mediante los documentos de transporte o embargo (MSDS) donde se encuentra el número ONU mediante el cual, se obtienen los datos de peligro, el nombre del “Mat-Pel” y la guía de emergencia donde se encuentran los riesgos potenciales, las medidas de seguridad pública y las acciones de respuesta ante una emergencia (1) y, de esta forma, hacer efectivas las decisiones asertivas en el menor tiempo posible para controlar el siniestro y evitar accidentes de mayor envergadura, pero la identificación mediante los documentos de transporte no es una opción permitida en una fuga o derrame de un “Mat-Pel”, ya que se encuentra en la cabina de conducción del vehículo de transporte y el nivel Advertencia tiene prohibido acercarse a la zona del siniestro

por medidas de seguridad, entonces la forma más segura de identificación es mediante el Panel Naranja que se encuentra ubicada en diversas posiciones del vehículo donde, además, se encuentra el número ONU. Sin embargo, ocurren situaciones en las que no pueden observar con claridad el Panel Naranja, ya que el nivel Advertencia se ubica en una posición segura fuera del área de fuga o derrame por el riesgo de exposición ante la fuga del “Mat-Pel”, entonces utilizan binoculares para poder visualizar el Panel Naranja lo cual les disminuye el tiempo de respuesta ante el siniestro.

Entonces, el problema de inseguridad a la que se encuentran expuestos el personal de bomberos incita al desarrollo de una solución tecnológica para evitar exponer su integridad al realizar estas tareas en ese entorno de alto riesgo, cumpliendo sus actividades del día a día.

En este caso, se propone el diseño aplicativo para la interpretación del Panel Naranja, el cual se encuentra ubicado en distintas partes de los vehículos o medios de transporte que acarrean materiales peligrosos, mediante procesamiento de imágenes, desde una distancia segura para la obtención rápida de los datos necesarios para una intervención posterior del nivel Operaciones, implementado en un “VANT” que cuenta con la facilidad de movilidad vía aérea para la obtención de las imágenes del Panel Naranja, para su posterior interpretación y que será del uso del nivel Advertencia del Cuerpo General de Bomberos de la región Junín.

### **1.1.1 Problema general**

¿Cómo realizar el diseño de un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín?

### **1.1.2 Problemas específicos**

- ¿Cómo se programará la interpretación del Panel Naranja?
- ¿Cómo se validará el funcionamiento de un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín?

## **1.2 Objetivos**

### **1.2.1 Objetivo general**

Realizar el diseño de un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín.

### **1.2.2 Objetivos específicos**

- Programar la interpretación del Panel Naranja.
- Validar el funcionamiento de un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín.

## **1.3 Justificación**

### **1.3.1 Justificación social**

La fuga de un “Mat-Pel” puede llegar a ser nocivo para la sociedad si no se controla el siniestro a tiempo, porque pueden llegar a una deflagración o tener contenidos tóxicos o incluso corrosivos que afecte a las personas que se encuentran cerca del siniestro. Entonces, desde que inicia el siniestro es esencial que el nivel Advertencia del Cuerpo General de Bomberos pueda reconocer el “Mat-Pel” con la mayor rapidez posible para identificar la guía de emergencia respectiva a usar y así disponer de todas las medidas necesarias para salvaguardar su integridad y la de la población.

### **1.3.2 Justificación teórica**

Esta investigación aportará conocimiento sobre la aplicación del OCR, extrayendo así, un texto de una imagen o video haciendo uso del lenguaje de programación Python y sus distintas librerías como CV2, Pytesseract y Tkinter, además se usó HTML, CSS y JavaScript para el diseño de una versión online mediante una página web.

### **1.3.3 Justificación práctica**

Para el reconocimiento e interpretación del Panel Naranja se hará mediante la captura de una imagen en un video en vivo y la carga de esta

imagen a una interfaz en Python que brindará los datos como el nombre del “Mat-Pel”, consejos de intervención, área de aislamiento, además de la guía de emergencia que contiene los riesgos potenciales, las medidas de seguridad pública y las acciones de respuesta ante una emergencia.

#### **1.3.4 Justificación metodológica**

Esta investigación utilizó una adaptación de la metodología VDI 2206 que consta de 4 fases las cuales son: fase de especificaciones, fase funcional, fase de diseño y fase de codificación para el desarrollo de un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1 Antecedentes del problema**

##### **2.1.1 Internacionales**

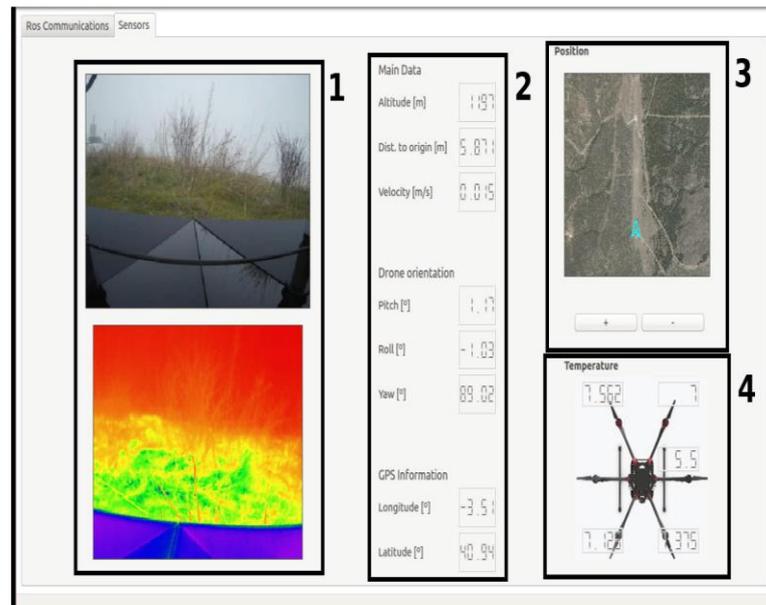
*“Análisis de imágenes multiespectrales adquiridas con vehículos aéreos no tripulados”* (3). Este artículo científico consistió en el uso de una VANT para la fotogrametría multiespectral, exactamente en el mapeo de sembríos empíricos de caña de azúcar. Se evaluó el uso de dos programas el Pix4D y *Agisoft Photoscan*, estos facilitan el tiempo y calidad de procesamiento de imágenes captadas por la cámara-sensor Parrot Sequoia el cual tiene la función principal de capturar imágenes multiespectrales. El programa más eficiente para este trabajo fue el Pix4D, este programa realiza correcciones radiométricas características a los ortomosaicos como se observa en la figura 2 (producto de imagen fotogramétricamente ortorrectificado organizado como mosaico a partir de una colección de imágenes). La resolución que se consiguió con la cámara implementada es de 11 cm/píxel, volando a 120 m de altura, lo cual permitió identificar problemas y tomar medidas específicas dependiendo de las áreas evaluadas. El VANT que se utilizó para esta investigación fue el 3DR RTF X8+.



**Figura 2. Ortomosaicos RGB construido con Pix4D volando a 120 metros de altura con 4 puntos de control foto insertados en los extremos de las flechas azules (3)**

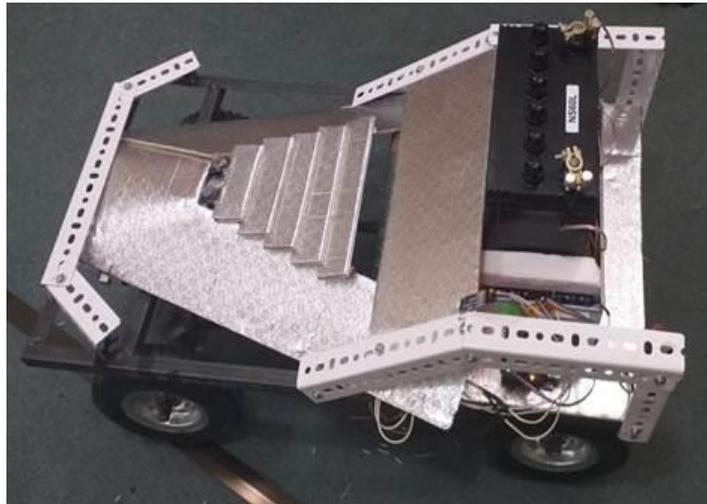
“Vehículo aéreo no tripulado para vigilancia y monitorización de incendios” (4). En este trabajo se presentó un UAV, implementado con sensores térmicos y ópticos, con sistema de procesamiento de datos y comunicación, con la finalidad de realizar tareas de supervisión y adquisición de datos provenientes de incendios en lugares rurales. Se recogió datos durante seis meses en colaboración de Dronitec S. L. y Divisek Systems que son empresas de telefónica digital de España, para poner en marcha un UAV con capacidad de supervisar y monitorizar incendios forestales transmitiendo información en tiempo real hacia los operadores todo esto gracias a la tecnología 4G. El UAV está implementado con una batería LiPo de 6 celdas y 12000 mAh el cual brinda una autonomía de 18 minutos, el alcance de vuelo es limitado por un controlador que permite trabajar en un área local de 3,5 km de radio a 7 km/h, cuenta con una cámara monocular RGB deportiva apto para adquirir imágenes y videos con resolución de 1920\*1080 a 60 fps en HD, también está implementada con una cámara térmica FLIR AX5 la cual se encarga de procesar imágenes térmicas del incendio, estas dos cámaras trabajando en conjunto arrojan datos de la temperatura en los focos de interés, se usaron sensores digitales de temperatura DS18B20 estos recogen la temperatura en un rango de 55 °C a 125 °C con precisión de  $\pm 0.5$  °C. Todos estos componentes están implementados en un

hexacóptero como se observa en la figura 3, basado en el chasis s900 de la empresa DJI, con una dimensión de 1250 mm \* 1250 mm \* 730 mm.



**Figura 3. Interfaz de la aplicación (4)**

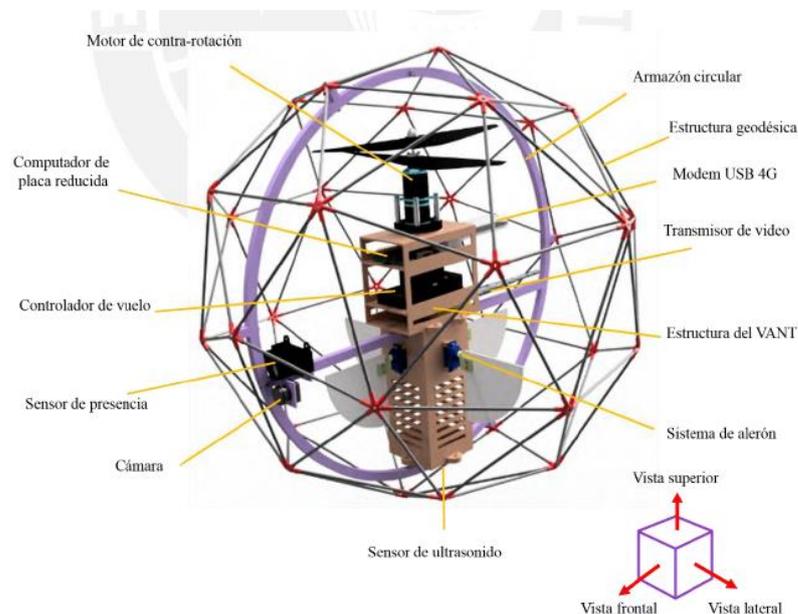
“Robot minero: sistema detector de gases utilizando sensores en tiempo real MIN – SIS 1.0 SDG-STR” (5). Este artículo científico consistió en diseñar un prototipo mecatrónico móvil, llamado Robot minero Min – SIS 1.0 SDG-STR. El prototipo está basado en la estructura de un auto, el cual se controla por un mando bluetooth teniendo la facultad de mandar imágenes por video, con la finalidad de obtener mediciones precisas de la densidad de diferentes gases tóxicos esto a través de sensores. Para el sistema de control se usó la tecnología Arduino el cual permite las conexiones de los motores (Servo motores), los sensores de gas y el módulo bluetooth, todo esto manipulado desde un celular para su transmisión y obtención de datos en vivo como se observa en la figura 4. Los sensores de gas (MQ-7) que se usaron son configurados de manera analógica permitiendo conocer la intensidad de los gases que pueden alojarse dentro de la mina.



**Figura 4. Diseño final del robot minero (5)**

### 2.1.2 Nacionales

Vehículo aéreo no tripulado para vigilancia en ambientes cerrados con detección de personas y obstáculos a su alrededor (6). La investigación residió en el diseño de un VANT que fue de uso en seguridad y vigilancia en ambientes cerrados. La metodología usada para el diseño la VDI 2221. El diseño incluye una estructura que impide los daños por colisión mientras se vuela en la figura 5. La vigilancia es proporcionada mediante una cámara proveída con visión nocturna y la transmisión de video es mediante una frecuencia baja de esta forma la señal pueda traspasar pisos y paredes en un radio de 50 m<sup>2</sup> a la redonda.



**Figura 5. Diseño del VANT (6)**

*“Diseño e implementación de robot terrestre teleoperado aplicando imágenes térmicas para mitigación de fuego operado por bomberos de La Victoria N.º 8” (7).* Este trabajo de investigación consistió en el diseño un robot móvil terrestre teleoperado con la función de percibir puntos de temperatura alta con la aplicación de imágenes térmicas. Este robot también será usado durante la mitigación de incendios, ya que contará con un sistema de dispersión de agua logrando resultados eficaces rápidamente. El diseño robusto que se presenta brinda las condiciones necesarias para poder adaptarse a entornos hostiles como se observa en la figura 6. Además, gracias a su sistema de detección de puntos elevados de temperatura, el robot muestra al operario un interfaz bastante sencillo y amigable que es donde se debe concentrar para poder controlar el incendio eficientemente. La adquisición de las imágenes térmicas se obtuvo con la instalación de cámaras infrarrojas en el robot, la alimentación eléctrica se hace mediante una batería SLI el cual mayormente es usado en los carros, el manejo del robot se hace mediante un joystick. El robot por ser robusto y contar con un sistema de dispersión de agua es muy pesado y al momento de desplazarse es bastante lento.



**Figura 6. Sistema integrado y operativo (7)**

*“Diseño e implementación de un robot móvil basado en tecnologías de teleoperación open-source para la toma de acciones de seguridad preventivas e inmediatas en caso de accidentes en minas subterráneas” (8).* Se desarrolló una solución usando la tecnología de programación *open-source*. Esto ayuda a la exploración y detección de distintos agentes químicos que pueden ser peligrosos dentro en las minas de tajo cerrado. El diseño mecánico y el sistema electrónico están implementados en un sistema robusto y seguro, capaces de resistir colisiones y explosiones pequeñas como se observa en la figura 7. Asimismo, la teleoperación es simple, atractivo y sencillo para el usuario. Los puntos más resaltantes fue la duración de energía que demostró un tiempo de funcionamiento de 40 min a 60 % de potencia que es un aproximado de 400 metros desplazados, tiene un alcance inalámbrico de hasta 80 metros pudiendo ser más en lugares libres, para su transmisión señales inalámbricas y obtención de los resultados que envían los distintos sensores se usa una red de punto a punto Xbee, su sistema de tracción oruga puede subir pendientes de hasta 30° de elevación. Este trabajo de investigación necesita estar conectado a internet para poder enviar todos los datos recolectados, el cual es un inconveniente ya que en interior de mina no se cuenta con el servicio de Internet.



**Figura 7. Presentación final del sistema robótico (8)**

## **2.2 Bases teóricas**

### **2.2.1 Visión artificial**

La visión artificial es considerada parte de la inteligencia artificial, es un grupo de métodos, teorías y técnicas que permiten simular el proceso de visión de los humanos con la capacidad de analizar y extraer de manera automática información de las imágenes. Además, permite crear aplicaciones y algoritmos para obtener información de objetos 3D a partir de imágenes obtenidas en 2D.

### **2.2.2 Software de procesamiento de imágenes**

Es un conjunto de herramientas que analizan las imágenes capturadas y extraen datos programados en un algoritmo que es compatible con los mismos

Para el procesamiento de imágenes se requieren capturar imágenes que cumplan características previamente especificadas manualmente dentro de la programación del software.

### **2.2.3 OpenCV**

Es una biblioteca de visión artificial desarrollada por la empresa Intel. Desde que se creó es utilizado en un sinnúmero de aplicaciones de procesamiento de imágenes. Entre sus principales características se encuentra el uso libre del software y su capacidad de multiplataforma que le permite trabajar con diferentes sistemas operativos.

### **2.2.4 Python**

Es un lenguaje de programación de código abierto, con la principal característica de adaptarse al estilo de programación de cada usuario porque es considerado un lenguaje de programación multiparadigma. Python es considerado uno de los lenguajes de programación con mayor compatibilidad con OpenCV, por la variedad de librerías que hacen posible el desarrollo del procesamiento de imágenes (9).

#### **2.2.4.1 Variables en Python**

Una variable es un objeto que puede variar de valor durante la ejecución de un programa, el nombre de una variable se define como identificador (9).

#### **2.2.4.2 Librerías en Python**

Una librería es un grupo de algoritmos funcionales codificados en Python que permiten un uso más práctico de ciertos programas al momento de crear o diseñar la programación deseada (9).

#### **2.2.4.3 Librería Tkinter**

Tkinter es una de las librerías con mayor uso en interfaces gráficas, que permite crear y modificar ventanas utilizando Python.

#### **2.2.4.4 Librería Pytesseract**

Es una librería que utiliza el OCR que permite extraer texto de una imagen o video digitalizando la imagen para su posterior reconocimiento.

#### **2.2.4.5 Funciones en Python**

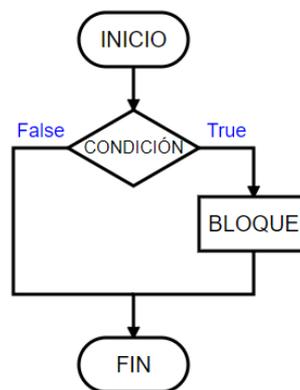
Una función es un bloque de códigos que se puede reutilizar múltiples veces sin necesidad de replicar el código, el uso de estas funciones facilita la legibilidad del código y evita errores al momento de replicar una parte del código.

En Python se crea la función utilizando el término “def” acompañado del nombre de la función y si se desea extraer datos de la función se utiliza el término “return” acompañado de la variable a extraer (10).

### 2.2.4.6 Sentencias condicionales

- **Condicional IF**

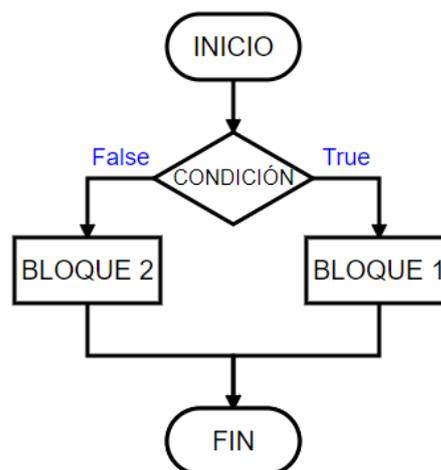
Permite ejecutar un programa (bloque) si la condición propuesta es verdadera, en caso contrario omite el programa como se puede observar en la figura 8.



*Figura 8. Diagrama de flujo condicional IF*

- **Condicional IF – ELSE**

Si la condición propuesta es verdadera ejecuta el programa del bloque 1 y si es falsa ejecuta el programa del bloque 2 como se puede observar en la figura 9.

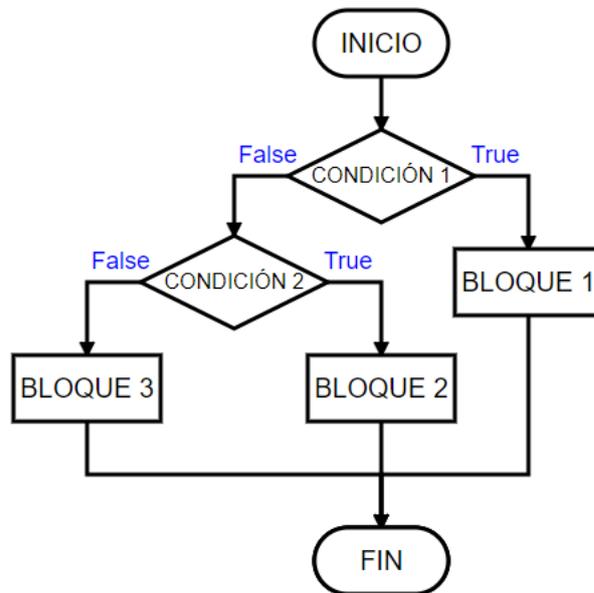


*Figura 9. Diagrama de flujo condicional IF – ELSE*

- **Condicional IF – ELIF - ELSE**

El orden es esencial en este condicional, si la primera condición es verdadera se ejecuta el programa del bloque 1 y si es

falsa revisa la segunda condición, si es verdadera se ejecuta el bloque 2 y si es falsa se ejecuta el bloque 3 como se puede visualizar en la figura 10.



**Figura 10. Diagrama de flujo condicional IF – ELIF – ELSE**

### 2.2.5 Estudio de tecnologías

Es muy conveniente tener presente las distintas alternativas cuando se está iniciando un desarrollo de una aplicación, ya sea web o móvil. En las siguientes líneas se van a presentar las diferentes tecnologías que se encuentran presentes en la web, se mostrarán sus ventajas y desventajas.

En el año 2021 los usuarios de móvil excedieron la mitad de la población mundial. Este fenómeno se ve reflejada en la concepción y uso de aplicaciones, la misma que da a conocer que la mayoría de las aplicaciones se encuentran en la *App store Google Play*.

El desarrollo de las aplicaciones se puede realizar de distintas maneras mediante las aplicaciones web, aplicaciones híbridas o nativas, las cuales tienes distintas configuraciones dependiendo de la aplicación que se desea crear.

### **2.2.6 Aplicaciones Web**

Las aplicaciones web no necesitan ser instaladas para poder ejecutarse. La mayor parte de su implementación está hecha en HTML, CSS Y JS, y se desempeñan en un navegador. Estas aplicaciones tienen la finalidad de interactuar con el dispositivo y el usuario (11).

#### **Las ventajas de las aplicaciones web**

- Puede acceder a la información del dispositivo para realizar ejecuciones complejas.
- Su desarrollo, distribución y sus pruebas son sencillas.
- Usa los estándares de la web.
- La mayoría de las industrias emplean esta tecnología, por lo que están sacando constantemente soporte para este tipo de aplicaciones.

#### **Desventajas**

- Es necesario contar con un navegador para poder dar soporte a esta aplicación.
- Tiene un rendimiento poco menor a las aplicaciones nativas, esto se da porque su ejecución es mediante en Java Script del navegador el cual tiene una potencia limitada.
- Se le dificulta acceder a componentes del dispositivo como la cámara o sensores.

### **2.2.7 HTML5**

Esta tecnología es un paso adelante el cual ayuda a mejorar los estándares de la web, también está impulsando el desarrollo en los dispositivos móviles. HTML5 da la posibilidad de brindar soporte a la mayoría de los navegadores, cuenta con un almacenamiento de datos en el navegador, teniendo así una base de datos (11).



*Figura 11 . Logotipo del estándar HTML 5. Tomada de Learnhtml*

### **2.2.8 CSS3**

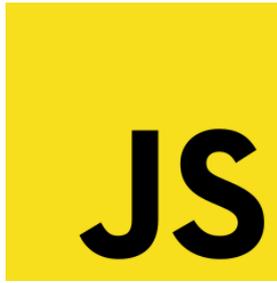
CSS (*Cascade Style Sheet*) que en español sería hoja de estilo en cascada, es un lenguaje diseñado para controlar la presentación de las páginas web, puede dar un estilo, como son los efectos visuales entre otros, a una estructura web desarrollada con HTML (12).



*Figura 12. Logotipo del estándar CSS 3. Tomada de Wikipedia*

### **2.2.9 Java Script**

JavaScript (JS) se trata de un lenguaje bastante popular en el mundo de programación ligero, compilado e interpretado justo a tiempo (*just in time*) que cuenta con funciones variadas de primera clase. Si bien es uno de los lenguajes más reconocidos de *scripting* (secuencias de comandos) usados en su mayoría para páginas web, también tiene un campo de aplicación en muchos otros entornos fuera del navegador, como lo es Node.js, CouchDB, Apache y Adobe Acrobat. Este lenguaje está basado en prototipos multiparadigma, dinámico, débilmente tipado, con un amplio fundamento para programación orientada a objetos, declarativo e imperativo (por ejemplo, programación funcional) (13).



*Figura 13. Logotipo de Java Script. Tomada de Wikipedia*

### **2.2.10 Aplicaciones híbridas**

Estas aplicaciones se realizan con la integración de tecnologías web, por lo que se puede deducir que son aplicaciones verdaderamente nativas, ya que se basan en WebView, y son desarrolladas dentro de un contenedor nativo, como resultado, tampoco están orientadas en web, esto se debe a que se empaquetan como aplicativos para distribución gracias a esto pueden acceder a los diferentes elementos del dispositivo, así como lo hacen las aplicaciones nativas (16).

El desarrollo de estas aplicaciones híbridas son prácticamente interminables esto se debe a que siempre están saliendo nuevos dispositivos, versiones de sistemas operativos e implementaciones de nuevas API, estos cambios hacen que se necesite estar pendiente del estado actual del software.

### **2.2.11 Diagrama de flujo**

Se denomina diagrama de flujo a la representación gráfica de un algoritmo o proceso (18). Se representa mediante la utilización de diversos símbolos como se puede apreciar en la figura 18.

Nombre	Símbolo	Función
<b>Terminal</b>		Representa el inicio y fin de un programa. También puede representar una parada o interrupción programada que sea necesaria realizar en un programa.
<b>Entrada / salida</b>		Cualquier tipo de introducción de datos en la memoria desde los periféricos o registro de información procesada en un periférico.
<b>Proceso</b>		Cualquier tipo de operación que pueda originar cambio de valor, formato o posición de la información almacenada en memoria, operaciones aritméticas, de transformaciones, etc.
<b>Decisión</b>		Indica operaciones lógicas o de comparación entre datos (normalmente dos) y en función del resultado de la misma determina (normalmente si y no) cual de los distintos caminos alternativos del programa se debe seguir
<b>Conector Misma Página</b>		Sirve para enlazar dos partes cualesquiera de un diagrama a través de un conector en la salida y otro conector en la entrada. Se refiere a la conexión en la misma pagina del diagrama
<b>Indicador de dirección o línea de flujo</b>		Indica el sentido de la ejecución de las operaciones
<b>Salida</b>		Se utiliza en ocasiones en lugar del símbolo de salida. El dibujo representa un pedazo de hoja. Es usado para mostrar datos o resultados.

**Figura 14. Simbología de diagrama de flujo (18)**

### 2.3 Definición de términos básicos

- **Número ONU:** número de identificación asignado a cada material peligroso por la Naciones Unidas
- **Número NIP:** número de identificación de peligro de un material peligroso
- **OpenCV:** biblioteca de código abierto para sistemas de procesamiento de imágenes
- **Mat-Pel:** siglas de “material peligroso”
- **GREE:** guía de respuesta en caso de emergencia
- **OCR:** reconocimiento óptico de caracteres (*optical character recognition*)

## **CAPÍTULO III**

### **METODOLOGÍA**

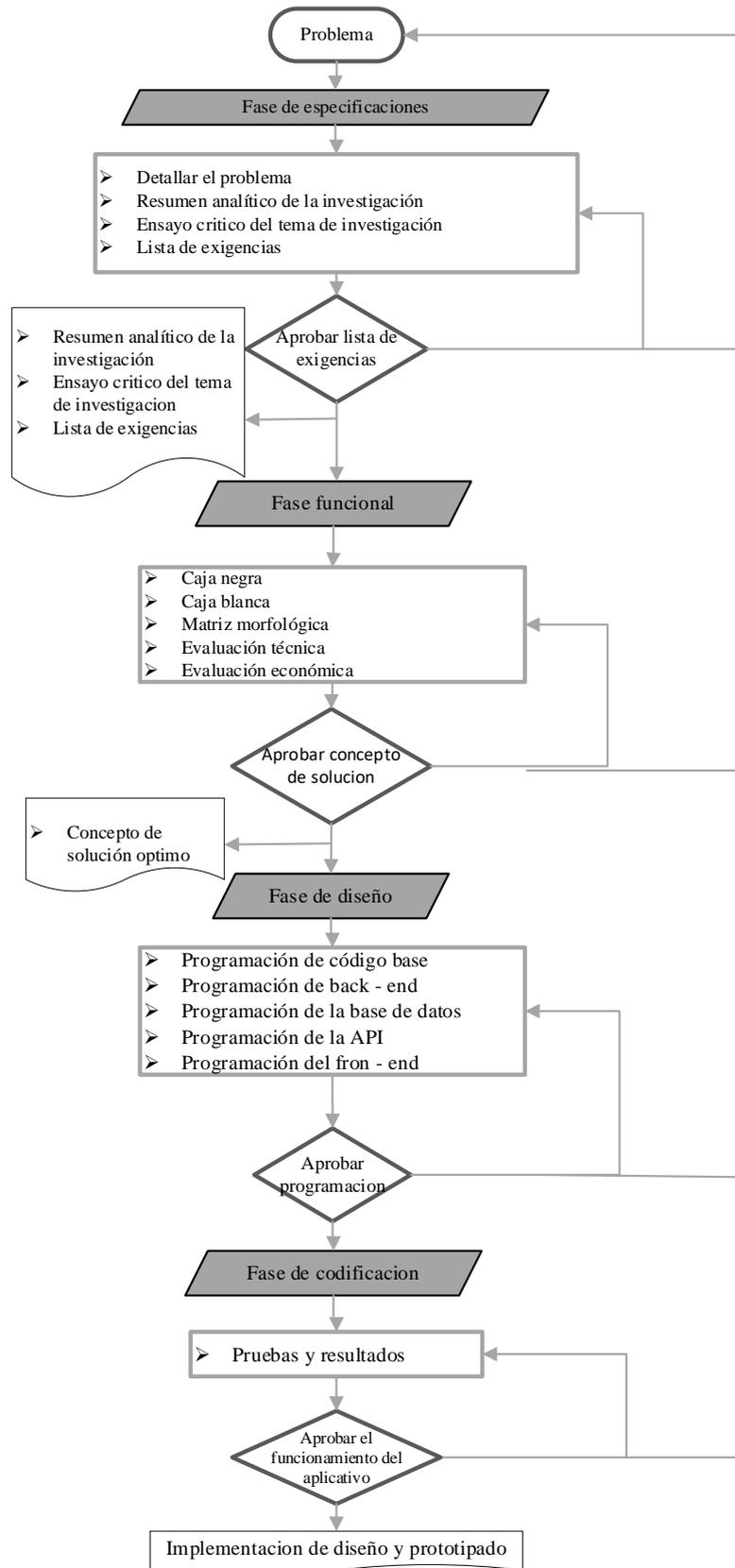
#### **3.1 Tipo de investigación**

El tipo de investigación es tecnológica porque emplea conocimiento obtenido y nuevos conocimientos que se adquieren en el proceso de investigación para brindar una solución a problemas de la sociedad (19).

Para solucionar el problema planteado, se emplearán conocimientos de programación en diferentes lenguajes para programar y diseñar un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” para el nivel Advertencia del Cuerpo General de Bomberos de la región Junín.

#### **3.2 Metodológica aplicada para el desarrollo de la solución**

La metodología utilizada será una adaptación de la VDI 2206 que contiene 4 fases las cuales son: fase de especificaciones, funcional, de diseño y de codificación como se puede apreciar en la figura 19.



**Figura 15. Esquema de la adaptación de la metodología VDI 2206**

## **Etapa 1**

### **Fase de especificaciones**

Esta fase se compone del análisis de problemática, requisitos y especificaciones.

Se deben instituir las exigencias para el diseño aplicativo como el tipo de software, investigaciones previas, alcances y limitaciones.

#### ➤ **Detallar el problema**

Resumen de la problemática en investigación.

#### ➤ **Resumen analítico de la investigación**

Artículos científicos de investigaciones previas.

**Tabla 1. Formato tabla de resumen analítico de la investigación**

Título	
Autores	
Año / lugar	
DOI	
Palabras clave	
Descripción general	
Fuentes	
Contenido	
Metodología	
Resultados-conclusiones	
Anexos	

*Nota:* disponible en formato de diseño de sistemas mecánicos UCCI 2021

#### ➤ **Ensayo crítico del tema de investigación**

Identificación del problema y posible solución.

**Tabla 2. Formato ensayo crítico del tema de investigación**

Introducción o planteamiento	Desarrollo o discusión	Conclusiones

*Nota:* disponible en formato de diseño de sistemas mecánicos UCCI 2021

#### ➤ **Lista de exigencias**

Lista de parámetros que tiene que cumplir el desarrollo del proyecto.

**Tabla 3. Formato de lista de exigencias**

Lista de exigencias		Universidad Continental	Pág.:
Proyecto		Ingeniería Mecatrónica	Fecha:
Características	Deseos o Exigencias	Condiciones	Responsable

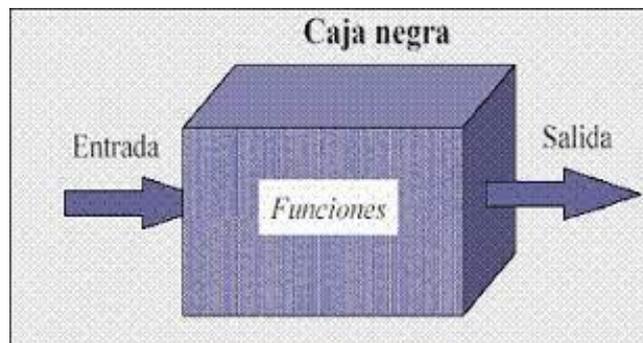
*Nota:* disponible en formato de diseño de sistemas mecánicos UCCI 2021

## Etapa 2

### Fase funcional

En esta etapa se debe formular una estructura de funciones que es interpretado mediante una caja negra. De esta forma se debe trazar la cadena de operaciones que se hará efecto en el desarrollo del diseño general.

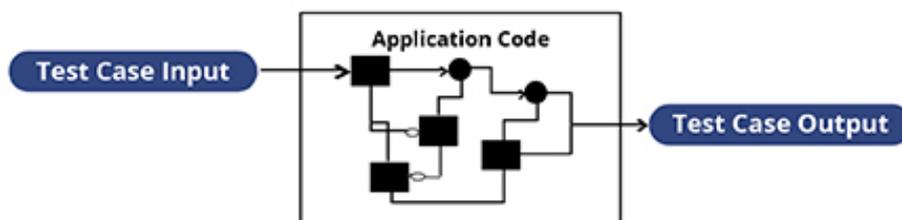
#### ➤ Caja negra



*Figura 16. Caja negra. Tomada de Teoría de los sistemas*

#### ➤ Caja blanca

### WHITE BOX TESTING APPROACH



*Figura 17. Caja blanca. Tomada de Software Testing Material*



### ✓ Evaluación económica

**Tabla 5. Formato de evaluación económica**

Diseño Mecatrónico		Valoración económica(yi)						Área de Diseño		
Proyecto										
p: Puntuación de 0 a 4 (Escala de valores según VDI 2206)										
0 = Muy bajo, 1 = Bajo, 2 = Medio, 3 = Regular, 4 = Perfecto (ideal)										
g: Peso ponderado de acuerdo a los criterios de evaluación										
Criterios de evaluación para diseños en fase de proyectos o conceptos										
Variantes de Concepto		Solución 1		Solución 2		Solución 3		Solución ideal		
		S1		S2		S3		S. Ideal		
N.º	Criterios de calificación	g	p	gp	p	gp	p	gp	p	gp
1										
2										
3										
4										
5										
Puntaje $\sum gp$									1.00	
Valor económico yi										

Nota: disponible en formato de diseño de sistemas mecánicos UCCI 2020

### ✓ Selección de solución óptima

Con los datos obtenidos de la evaluación técnica-económica se determina la solución más viable.

## Etapa 3

### Fase de diseño

#### ✓ Programación de código base

Definir funciones y estructura de la lectura OCR.

#### ✓ Programación del *back end*

Programación de la obtención de caracteres a través de OCR y su identificación.

#### ✓ Programación de la base de datos

Llenado de información necesitada para el código.

#### ✓ Programación del *front end*

Programación del aplicativo que se usará para la obtención de datos.

## **Etapa 4**

### **Fase de codificación**

Implementación del diseño y prototipado.

En esta etapa se debe implementar los componentes del diseño respetando la estructura de funciones para el prototipado y la posterior validación de las exigencias.

## **CAPÍTULO IV**

### **ANÁLISIS Y DISEÑO DE LA SOLUCIÓN**

#### **4.1. Identificación de los requerimientos**

Consiste en desarrollar la fase de especificaciones de la metodología VDI 2206 adaptada.

##### **4.1.1 Detallar el problema**

La inseguridad a la que se expone el cuerpo general de bomberos al laborar con materiales peligrosos “Mat-Pel” cuando se suscitan fugas o derrames de estos “Mat-Pel”, amerita plantear soluciones tecnológicas para evitar accidentes en el proceso de identificación del “Mat-Pel”, el cual lo lleva a cabo el nivel Advertencia del Cuerpo General de Bomberos.

Una de las formas de identificar los “Mat-Pel” es mediante el Panel Naranja en el cual se encuentra el número ONU, a partir de la identificación del número ONU se puede determinar el nombre del “Mat-Pel” y posterior a eso obtener la guía de emergencia asignada a partir del nombre del “Mat-Pel” en la GRE2020, en esta guía se detallan todos los riesgos potenciales, las medidas para la seguridad pública y las acciones de respuesta a emergencia (1).

La problemática surge en que para visualizar el Panel Naranja el cuerpo de bomberos tiene que hacerlo desde una distancia segura para

evitar accidentes y, en muchos casos, la visualización no es clara porque hacen uso de binoculares además que una vez visualizado el número ONU ellos tienen que buscar la guía de emergencia de manera manual y el tiempo para poder obtener todos los datos es crucial, porque en cualquier momento puede suscitar algún accidente de mayor consideración.

#### 4.1.2 Resumen analítico de la investigación

**Tabla 6. Análisis de imágenes multiespectrales adquiridas con vehículos aéreos no tripulados**

Título	Análisis de imágenes multiespectrales adquiridas con vehículos aéreos no tripulados
<b>Autores</b>	Samy Kharuf-Gutiérrez, Luis Hernández-Santana, Rubén Orozco-Morales, Osmany de la C. Aday Díaz, Irenaldo Delgado Mora
<b>Año / lugar</b>	2018/ Santa Clara - Cuba
<b>DOI</b>	<a href="http://scielo.sld.cu/scielo.php?script=sci_arttext&amp;pid=S1815-59282018000200007&amp;lang=es">http://scielo.sld.cu/scielo.php?script=sci_arttext&amp;pid=S1815-59282018000200007&amp;lang=es</a>
<b>Palabras claves</b>	Agricultura de precisión; índices vegetativos; procesamiento de imágenes; Pix4D; Agisoft Photoscan.
<b>Descripción general</b>	Este artículo científico consistió en el uso de una VANT para la fotogrametría multiespectral, exactamente en el mapeo de sembríos empíricos de caña de azúcar. Se evaluó el uso de dos programas el Pix4D y Agisoft Photoscan, estos facilitan el tiempo y calidad de procesamiento de imágenes captadas por la cámara-sensor Parrot Sequoia el cual tiene la función principal de capturar imágenes multiespectrales.
<b>Fuentes</b>	Best S, Zamora I. Tecnologías aplicables en Agricultura de Precisión. Uso de tecnología de precisión en evaluación, diagnóstico y solución de problemas productivos. Santiago de Chile: Fundación para la Innovación Agraria, 2008.  García-Cervigón D., José J. Estudio de índices de vegetación a partir de imágenes aéreas tomadas desde UAS\RPAS y aplicaciones de estos a la agricultura de precisión: Universidad Complutense de Madrid 2015. Disponible en: <a href="http://eprints.ucm.es/31423/1/TFM_Juan_Diaz_Cervignon.pdf">http://eprints.ucm.es/31423/1/TFM_Juan_Diaz_Cervignon.pdf</a>  García-Ferrer A. Potencialidad y nuevas orientaciones en el uso de la teledetección en agricultura y selvicultura. Ambiental: la revista del Ministerio de Medio Ambiente. 2013(105):6-15.
<b>Contenido</b>	El programa más eficiente para este trabajo fue el Pix4D, este programa realiza correcciones radiométricas características a los ortomosaicos (producto de imagen fotogramétricamente ortorrectificado organizado como mosaico a partir de una colección de imágenes). La resolución que se consiguió con la cámara implementada es de 11 cm/pixel, volando a 120 m de altura, lo cual permitió identificar problemas y tomar medidas específicas dependiendo de las áreas evaluadas. El VANT que se utilizó para esta investigación fue el 3DR RTF X8+.
<b>Metodología</b>	VDI 2206
<b>Resultados - conclusiones</b>	Las pruebas arrojaron que no se puede conseguir imágenes ortomosaicos RGB, sin huecos ni distorsiones, manteniendo un vuelo de 40 m y con una velocidad de 6 m / seg usando el sensor

Sequoia. Por otro lado, la cámara Parrot Sequoia se encargó de conseguir imágenes multiespectrales se obtuvieron resultados positivos. Se trabajo con esa configuración de la cámara ya que la banda RGB es solo un valor agregado, ya que en esta configuración se pudo obtener una mejor calidad de imágenes multiespectrales. Se evidenció que el programa que mejor se adaptó a este trabajo por su flexibilidad y su mayor potencia al momento de arrojar resultados más exactos fue el software Pix4D.

---



#### Anexos

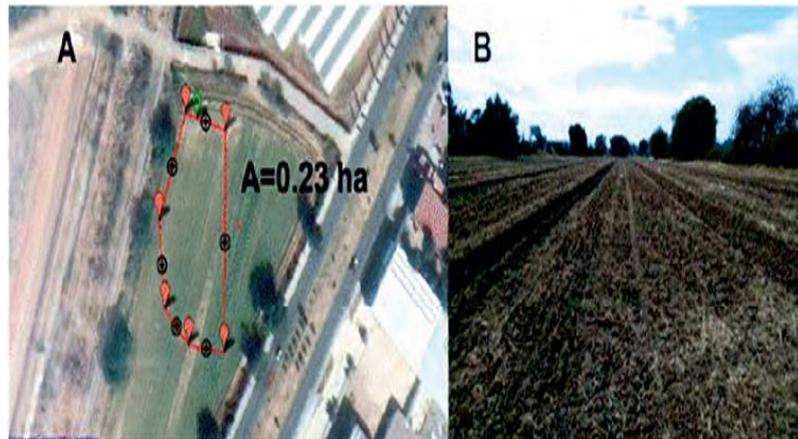


**Tabla 7. Evaluación de un prototipo de RPAS para el levantamiento topográfico con imágenes RGB**

Título	Evaluación de un prototipo de RPAS para el levantamiento topográfico con imágenes RGB
<b>Autores</b>	Juan J. Pérez Paredes, Gilberto J. López Canteñs, Noé Velázquez López
<b>Año/ Lugar</b>	2021/ Texcoco-México
<b>DOI</b>	<a href="https://www.redalyc.org/journal/5862/586266250004/">https://www.redalyc.org/journal/5862/586266250004/</a>
<b>Palabras Clave</b>	dron, topografía, fotogrametría, georreferenciación, puntos de control en tierra.
<b>Descripción General</b>	<p>Este artículo científico consistió en la evaluación de un prototipo de RPAS que tenía como principal función el levantamiento topográfico, partiendo de imágenes RGB georreferenciadas, este incluyó puntos de control ubicados en la tierra. El campo experimental fue la Universidad Autónoma Chapingo, México.</p> <p>Claros, R. A.; Guevara, A. E.; Pacas, N. R.: <i>Aplicación de fotogrametría aérea en levantamientos topográficos mediante el uso de vehículos aéreos no tripulados, [en línea]</i>, Universidad de El Salvador, Tesis de licenciatura, El Salvador, p. 319 2016, <i>Disponible en:</i> <a href="http://ri.ues.edu.sv/id/eprint/14218/1/50108282.pdf">http://ri.ues.edu.sv/id/eprint/14218/1/50108282.pdf</a>, [Consulta: 17 de junio de 2020].</p>
<b>Fuentes</b>	<p>Florentino, R. C.: <i>Aplicación de fotogrametría con RPAS para mejorar la efectividad en cuantificación de la explotación en la cantera Santa Genoveva, [en línea]</i>, Universidad César Vallejo, Tesis de licenciatura, Perú, p. 123, 2017, <i>Disponible en:</i> <a href="http://repositorio.ucv.edu.pe/handle/20.500.12692/23113">http://repositorio.ucv.edu.pe/handle/20.500.12692/23113</a></p> <p>Lorduy, S.: <i>Generación de ortoimágenes usando vehículos aéreos no tripulados aplicado a la agricultura, [en línea]</i>, Universidad Eafit, Tesis de licenciatura, Colombia, p. 80, 2017, <i>Disponible en:</i> <a href="https://repository.eafit.edu.co/handle/10784/12376">https://repository.eafit.edu.co/handle/10784/12376</a></p>
<b>Contenido</b>	<p>El prototipo que se usó fue el RPAS cuadricóptero, el cual se equipó con una cámara RGB Hawkeye Firefly 8SE de la mano con el sensor Sony IMX 117, el cual cuenta con una resolución de 4608 por 3456 pixeles, se usó una controladora de vuelo PixHawk, este utiliza un código abierto y se encarga de supervisar el completo funcionamiento del RPAS en sus diferentes modos de vuelo como también se encarga de la cámara y los giros del gimbal, asimismo tiene la capacidad de georreferenciar las imágenes capturadas por el RPAS.</p> <p>El software que se encargó de procesar las imágenes fue el PIX4D mapper, las imágenes que se le alimentaba al software se orientaban mediante la generación de puntos de enlace. La altura que alcanzaba el prototipo fue de 20 m, con una velocidad de 1 m/s llegando a cubrir un área de 2151 m<sup>2</sup>. Las imágenes capturadas fueron corregidas por el software PIX4D, se realizó una comparación entre el control en tierra y la nube densa de puntos para luego realizar un levantamiento topográfico GPS en postproceso, mediante este proceso su obtuvo unos errores absolutos y estándares de 0.020 y 0.024 m.</p>
<b>Metodología</b>	VDI 2206
<b>Resultados-Conclusiones</b>	El procedimiento se realizó en una parcela, el cual requería en levantamiento topográfico, el prototipo RPAZ se encargó de la nivelación de tierras de esta parcela. Se redujeron los errores en la altura geodésica, adjuntando puntos de control en tierra, consiguiendo así la generación de la nube densa y el modelo digital de elevación.

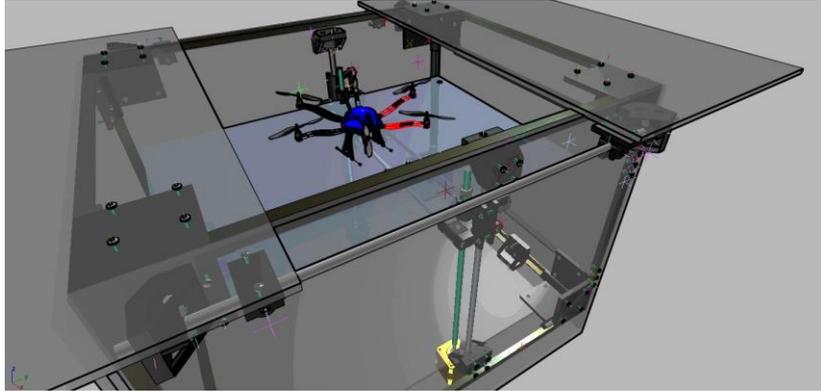
Se partió del análisis de los datos de levantamiento topográfico por medio del proceso de imágenes del prototipo RPA y el sistema GNSS, se sacaron errores absolutos que fueron inferiores al 0.033 m en el 85 % de la información, entonces se puede considerar que esta investigación cumplió con su objetivo y se podría considerar como una alternativa a fin de realizar levantamientos topográficos en campos agrícolas.

Anexos

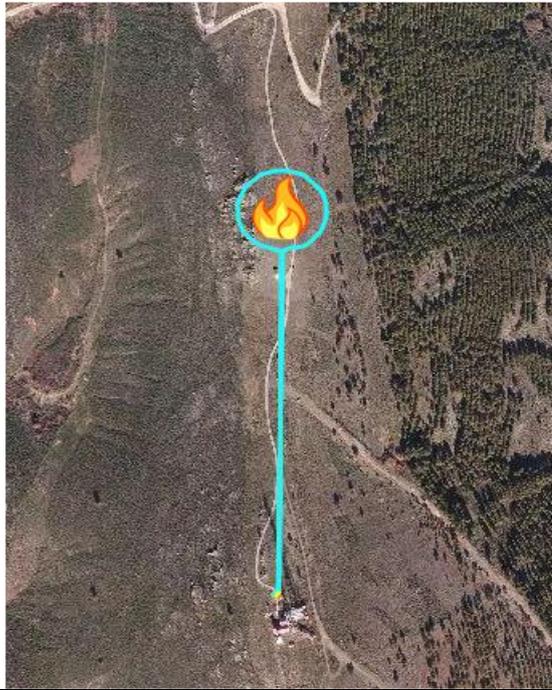


**Tabla 8. Vehículo aéreo no tripulado para vigilancia y monitorización de incendios**

<b>Título</b>	<b>Vehículo aéreo no tripulado para vigilancia y monitorización de incendios</b>
<b>Autores</b>	Madridano, A., Campos, S., Al-Kaff, A., García, F., Martín, D., Escalera, A.
<b>Año/ Lugar</b>	2020/ Madrid-España
<b>DOI</b>	<a href="https://doi.org/10.4995/riai.2020.11806">https://doi.org/10.4995/riai.2020.11806</a>
<b>Palabras Clave</b>	UAV; detección de incendios; sensores; vehículos autónomos inteligentes; navegación; vigilancia; monitorización.
<b>Descripción General</b>	En este trabajo se presentó un UAV, implementado con sensores térmicos y ópticos, con sistema de procesamiento de datos y comunicación, con la finalidad de realizar tareas de supervisión y adquisición de datos provenientes de incendios en lugares rurales.
<b>Fuentes</b>	<p>Pajares, G., 2015. Overview and status of remote sensing applications based on unmanned aerial vehicles (UAV). <i>Photogrammetric Engineering &amp; Remote Sensing</i> 81 (4), 281–330.</p> <p>Toriz, A., Raygoza, M., Martínez, D., 2017. Modelo de inclusión tecnológica UAV para la prevención de trabajos de alto riesgo, en industrias de la construcción basado en la metodología IVAS. <i>Revista Iberoamericana de Automática e Informática industrial</i> 14 (1), 94–103.</p> <p>Yuan, C., Liu, Z., Zhang, Y., 2016. Vision-based Forest fire detection in aerial images for firefighting using UAV. In: 2016 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, pp. 1200–1205.</p>
<b>Contenido</b>	El UAV está implementado con una batería LiPo de 6 celdas y 12000 mAh el cual brinda una autonomía de 18 minutos, el alcance de vuelo es limitado por un controlador que permite trabajar en un área local de 3,5 km de radio a 7 km/h, cuenta con una cámara monocular RGB deportiva apto de adquirir imágenes y videos con resolución de 1920*1080 a 60 fps en HD, también esta implementada con una cámara térmica FLIR AX5 la cual se encarga de procesar imágenes térmicas del incendio, estas dos cámaras trabajando en conjunto arrojan datos de la temperatura en los focos de interés, se usaron sensores digitales de temperatura DS18B20 estos recogen la temperatura en un rango de 55 °C a 125 °C con precisión de ± 0.5 °C. Todos estos componentes están implementados en un hexacóptero basado en el chasis s900 de la empresa DJI, con una dimensión de 1250 mm * 1250 mm * 730 mm.
<b>Metodología</b>	VDI 2206
<b>Resultados-Conclusiones</b>	Se recogió datos durante seis meses en colaboración de Dronitec S. L. y Divisek Systems que son empresas de telefónica digital de España, para poner en marcha un UAV con capacidad de supervisar y monitorizar incendios forestales transmitiendo información en tiempo real hacia los operadores, todo esto gracias a la tecnología 4G. Se pudo implementar esta nueva tecnología gracias a la interacción de lo que sería universidad-empresa complementando así la labor del cuerpo de emergencia priorizando la lucha contra el fuego.



**Anexos**



### 4.1.3 Ensayo crítico del tema de investigación

Tabla 9. Tema de investigación

Tema de Investigación		
Introducción o planteamiento	Desarrollo de la discusión	Conclusiones
<p>La inseguridad a la que se expone el nivel Advertencia del Cuerpo General de Bomberos al cumplir su función de identificar el “Mat-Pel” que se encuentra en el Panel Naranja, para poder informar las acciones necesarias a realizar de acuerdo a la Guía de Respuesta en Casos de Emergencia “GRE”.</p> <p>En estos casos la problemática surge en que para visualizar el Panel Naranja el cuerpo de bomberos tiene que hacerlo desde una distancia segura para evitar accidentes y en muchos casos la visualización no es clara, porque hacen uso de binoculares, además que una vez visualizado el número ONU ellos tienen que buscar la guía de emergencia de manera manual y el tiempo para poder obtener todos los datos es crucial porque en cualquier momento puede suscitarse algún accidente de mayor consideración.</p>	<p>El procesamiento de imágenes es una tecnología que tiene muy buenos resultados en otras áreas como el reconocimiento automático de matrículas, su uso se da cuando un automóvil llega a un punto de control, la aplicación obtiene la imagen de video y lo convierte en datos, dichos datos son procesados con algoritmos de vanguardia y lo comparan con una base de datos interna el cual te da una respuesta super rápida y te permite tomar una decisión (20).</p> <p>La institución de los bomberos constantemente se encuentra exponiendo sus vidas como en el suceso que ocurrió el 23 de enero del 2020 donde hubo 2 muertos y más de 40 heridos después de la deflagración de un camión cisterna que llevaba gas licuado de petróleo (GLP) que es considerado como un “Mat-Pel” de mucho riesgo (2).</p>	<p>Entonces, el problema de inseguridad a la que se encuentran expuestos el personal de bomberos incita al desarrollo de una solución tecnológica para evitar exponer su integridad en el proceso de identificación del “Mat-Pel”.</p> <p>En este caso, se propone el diseño aplicativo para la interpretación del Panel Naranja mediante procesamiento de imágenes (OCR), desde una distancia segura para la obtención rápida de los datos necesarios, para una intervención posterior del nivel Operaciones, implementado en un “VANT” que cuenta con la facilidad de movilidad vía aérea para una mejor obtención de las imágenes del Panel Naranja para su posterior interpretación donde se obtendrá el número ONU, nombre del “Mat-Pel” y la guía de emergencia asignada para el uso del nivel Advertencia del Cuerpo General de Bomberos de la región Junín.</p>

#### 4.1.4 Lista de exigencias

Tabla 10. *Lista de exigencias*

Lista de exigencias		Universidad Continental	Pág.
Proyecto		Ingeniería Mecatrónica	Fecha
Caract.	Deseos o Exigencias	Condiciones	Responsable
Función	E	Diseñar una aplicación para identificar el panel naranja.	Grupo de trabajo
Función	E	El diseño del aplicativo se puede acoplar a diferentes tipos de VANT.	Grupo de trabajo
Función	D	Se buscará que la aplicación tenga la capacidad de interpretar el Panel Naranja de manera precisa.	Grupo de trabajo
Función	E	El interfaz del aplicativo deberá de ser simple e intuitivo y con gran capacidad de interacción con el operador.	Grupo de trabajo
Función	E	La aplicación considerará un funcionamiento offline.	Grupo de trabajo
Función	E	El aplicativo poseerá opciones de personalización.	Grupo de trabajo
Función	E	La aplicación mantendrá una buena seguridad para la confidencialidad de los datos.	Grupo de trabajo
Función	E	El aplicativo se podrá actualizar cuando se requiera.	Grupo de trabajo
Función	E	La cámara deberá transferir las imágenes en tiempo real evitando interferencias.	Grupo de trabajo
Función	D	La cámara deberá contar con una buena calidad de recepción de imagen.	Grupo de trabajo

## 4.2 Análisis de la solución

Consiste en desarrollar la fase funcional de la metodología VDI 2206 adaptada.

### 4.2.1 Black-Box



#### Entradas

##### ✓ Señal visual de la cámara

Video en vivo captado mediante el transmisor de la cámara del VANT.

✓ **Inicio de la aplicación**

Establecer función de encendido emitida del operador hacia el VANT.

**Salidas**

✓ **Número ONU**

Es un numero de 4 dígitos asignado a cada material peligroso por la ONU.

✓ **Nombre del “Mat-Pel”**

Con el procesamiento de imagen el aplicativo da como respuesta el nombre del material peligroso.

✓ **Guía de emergencia por usar**

Con el nombre del material peligroso la aplicación da como respuesta un plan de contingencia específico para cada material peligroso.

**4.2.2 Caja blanca**

Esta estructura representa de manera grafica las funciones principales del aplicativo a diseñar como se puede visualizar en la figura 23, cumpliendo de esta forma los objetivos planteados.

### Diseño de un aplicativo para la interpretación del Panel Naranja, implementado en un "VANT"

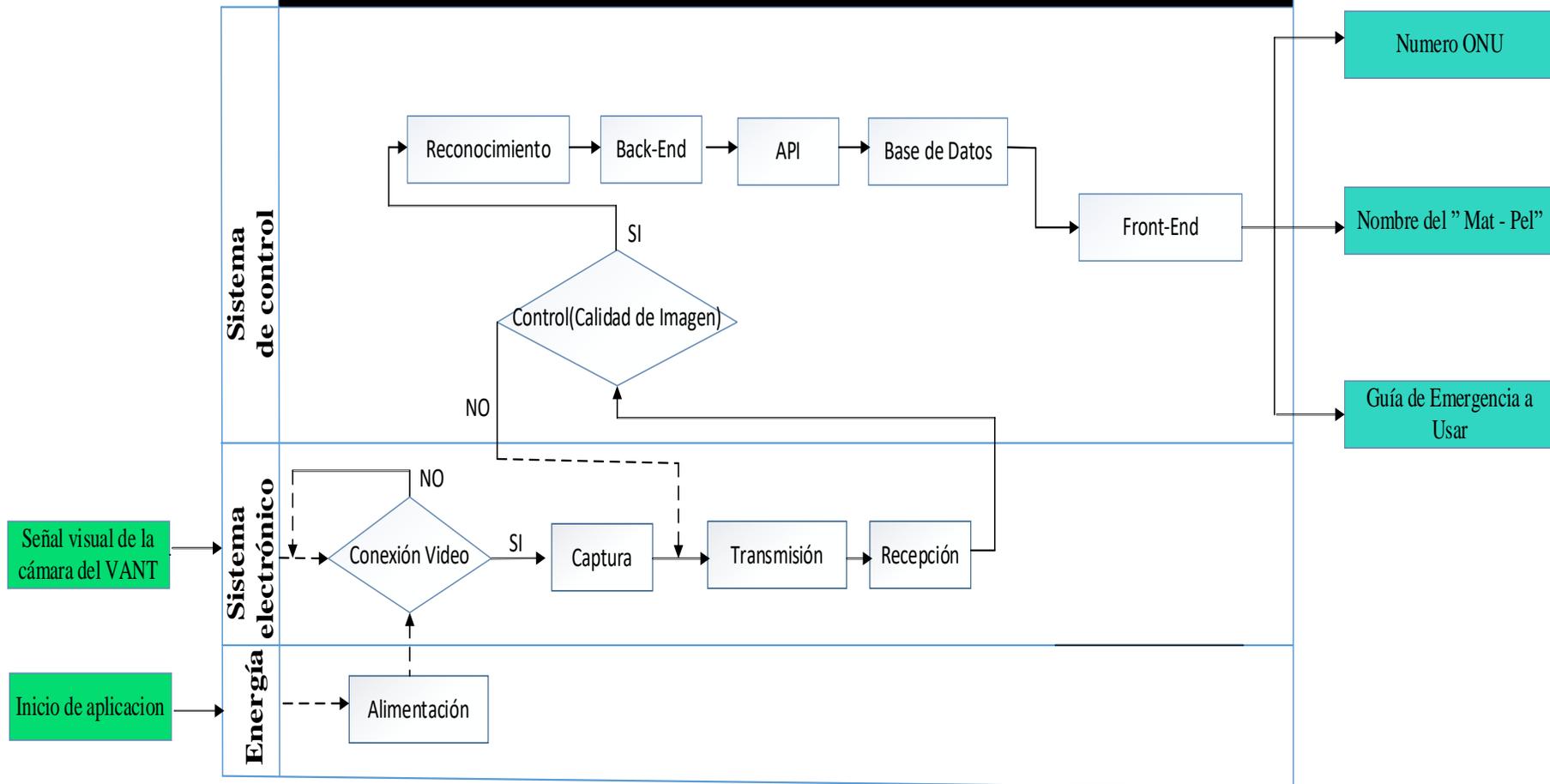


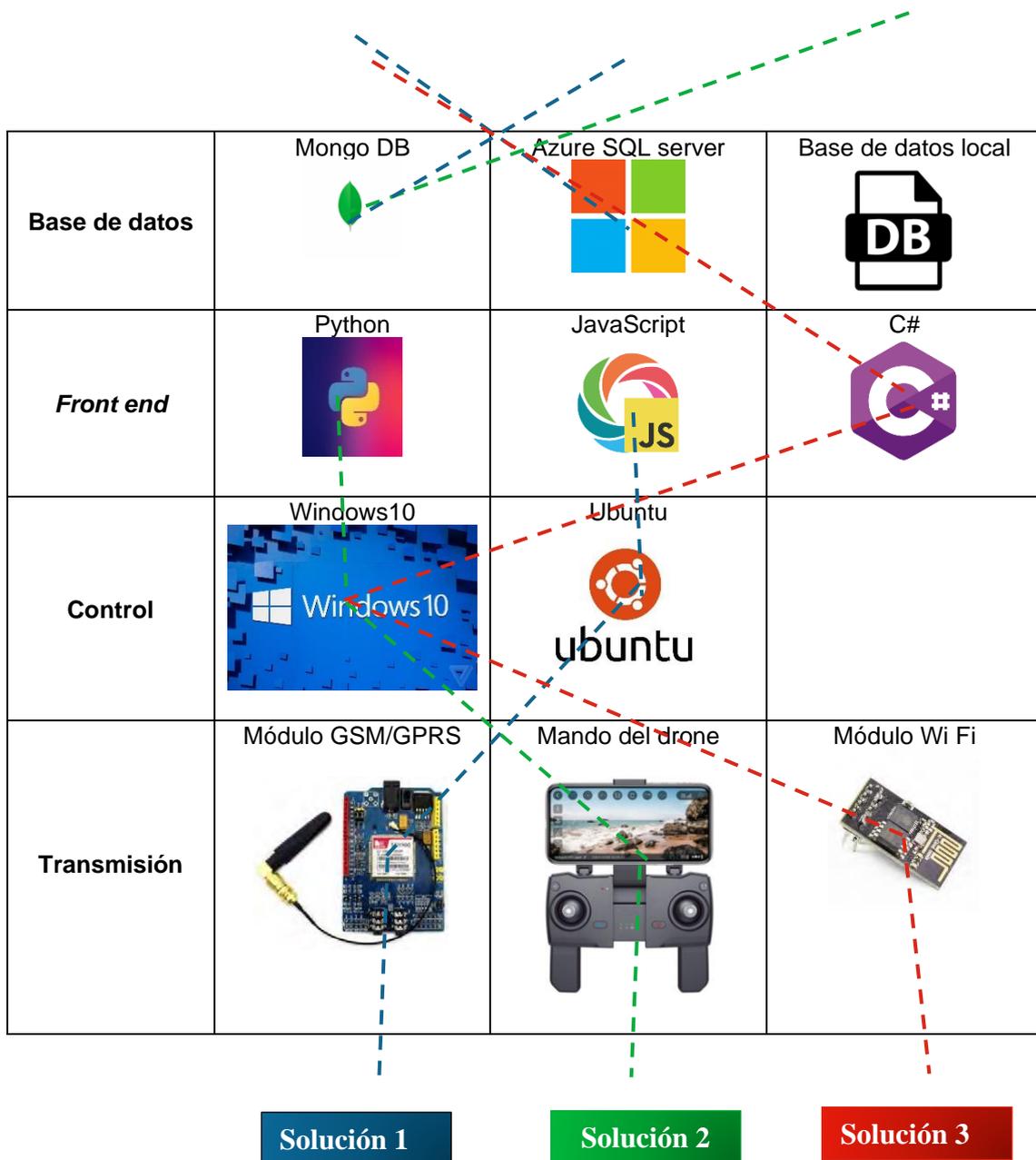
Figura 19. Caja blanca

### 4.2.3 Matriz morfológica

La matriz que se observa en la tabla 11 incluye las alternativas que tienen el propósito de hacer efectivas las funciones propuestas, de este modo se extraen las posibles soluciones, que pasaran a una posterior evaluación técnica-económica para localizar un diseño apropiado del aplicativo implementado en un VANT.

Tabla 11. *Matriz morfológica*

Sistema de vehículo aéreo no tripulado (VANT)			
Función parcial	Concepto solución 1	Concepto solución 2	Concepto solución 3
Alimentación	Cargador de laptop 	Fuente de energía 	
Captura	Camara FPV 	Cámara del VANT 	Camara multi espectral 
Recepción	Celular 	Laptop 	
Reconocimiento	Reconocimiento óptico de caracteres 		
Back end	Python 	Java Script 	C# 
API	WordPress 	HTML, CSS y JS 	



# Concepto de solución 1

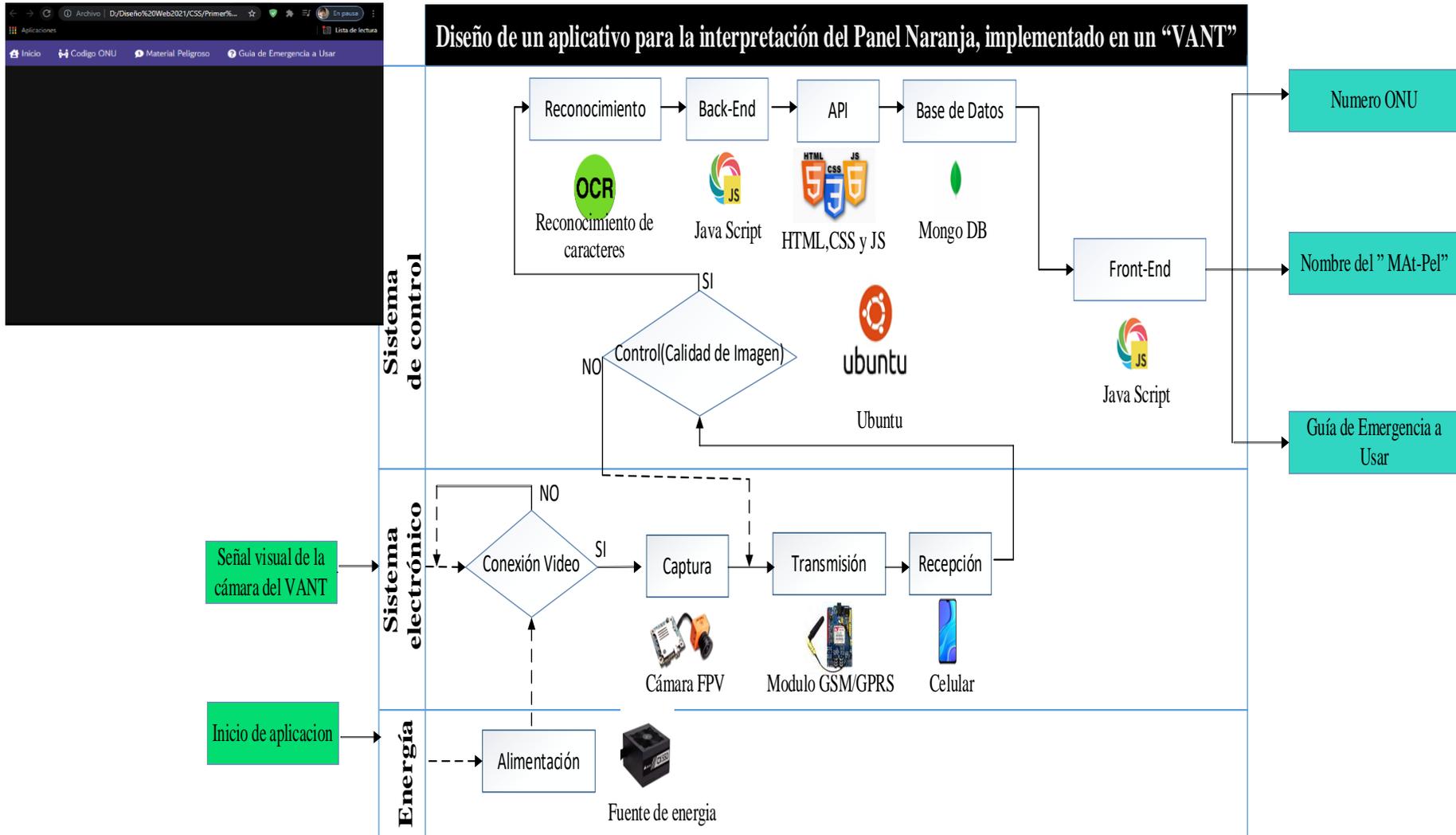


Figura 20. Concepto de solución 1

## Concepto de solución 2

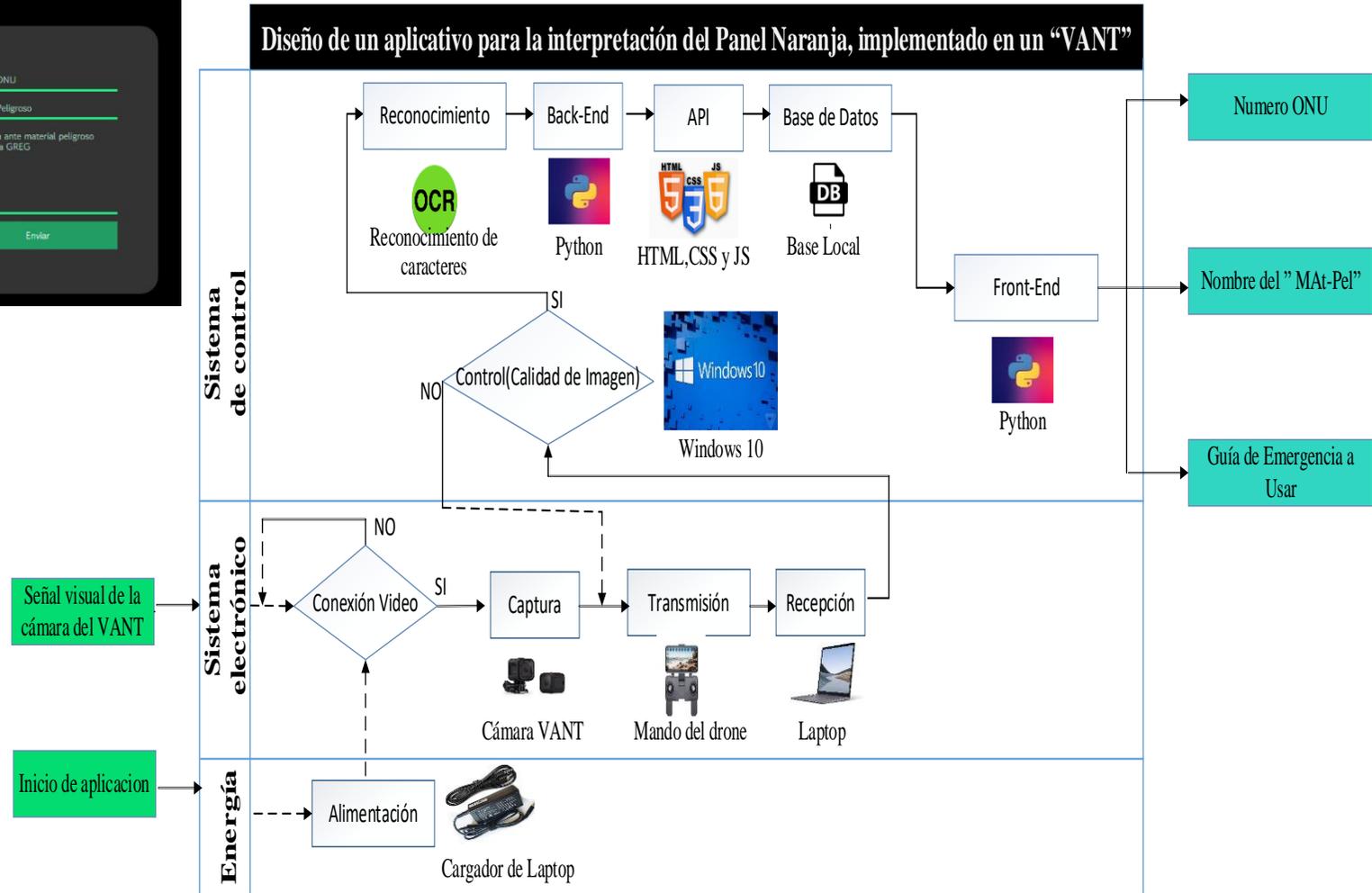


Figura 21. Concepto de solución 2

### Concepto de solución 3

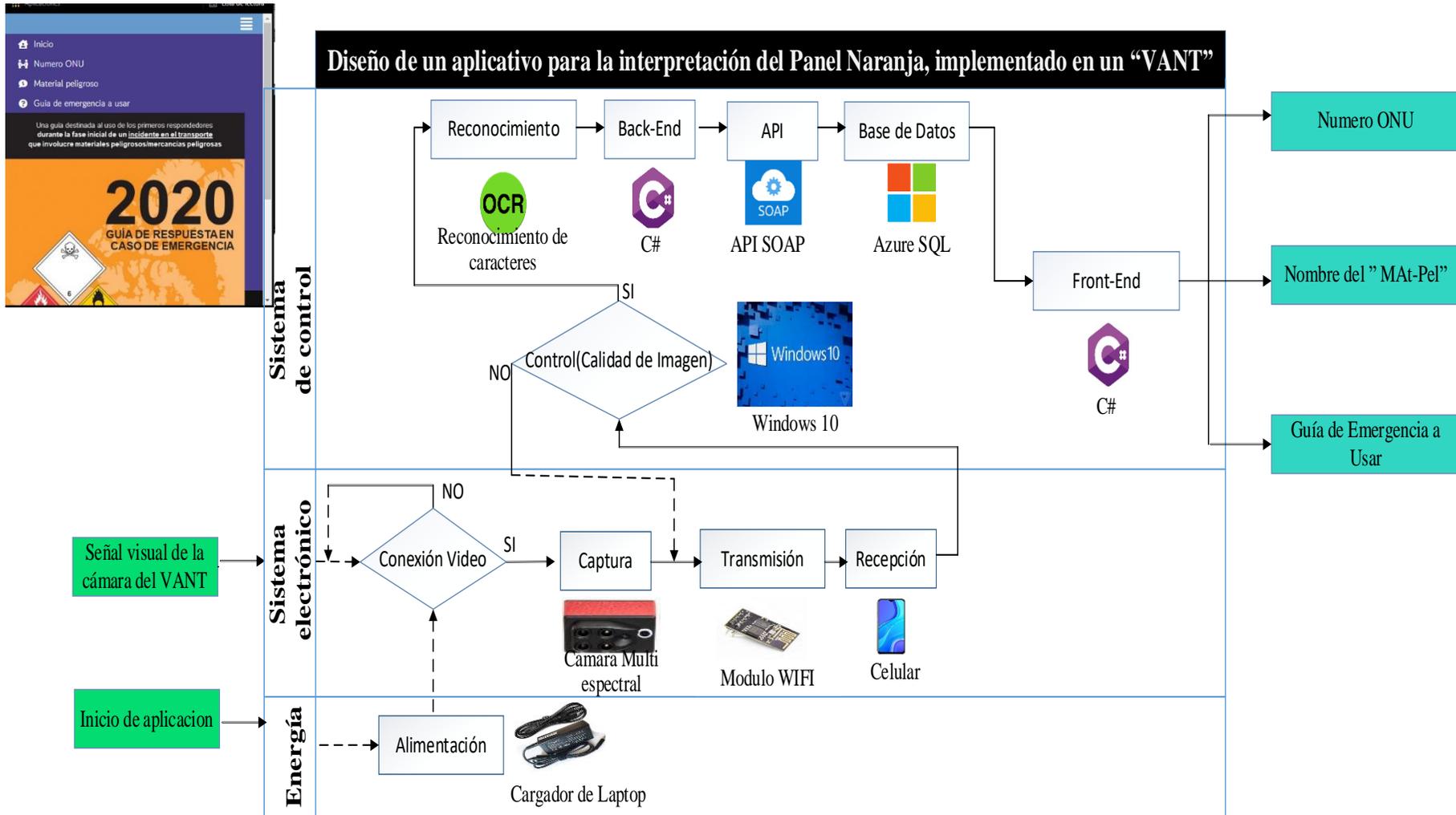


Figura 22. Concepto de solución 3

## 4.2.4 Evaluación técnico – económica del concepto de solución

### 4.2.4.1 Valoración técnica

**Tabla 12. Valoración técnica**

Diseño Mecatrónico		Valoración técnica (xi)								Área de diseño	
<b>Proyecto:</b> Aplicativo para la interpretación del Panel Naranja, implementado en un "VANT"											
p: Puntuación de 0 a 4 (Escala de valores según VDI 2206)											
0 = Muy bajo, 1 = Bajo, 2 = Medio, 3 = Regular, 4 = Perfecto (ideal)											
g: Peso ponderado de acuerdo a los criterios de evaluación											
Criterios de evaluación para diseños en fase de proyectos o conceptos											
Variantes de concepto		Solución 1		Solución 2		Solución 3		Solución ideal			
		S1	S2	S3	S ideal						
N.º	Criterios de calificación	g	p	gp	p	gp	p	gp	P	gp	
1	Funcionalidad	8	2	16	3	24	3	24	4	32	
2	Compatibilidad	6	2	12	3	18	3	18	4	24	
3	Autonomía	6	3	18	4	24	4	24	4	24	
4	Capacidad de datos	4	2	8	4	16	2	8	4	16	
5	Facilidad de operación	1	3	30	4	40	2	20	4	40	
6	Mantenimiento	5	3	15	3	15	2	10	4	20	
7	Rapidez	8	2	16	3	24	3	24	4	32	
8	Seguridad	8	1	8	2	16	2	16	4	32	
9	Portabilidad	8	1	8	2	16	3	24	4	32	
Puntaje $\sum gp$		131		193		150		252			
Valor Técnico xi		0.51		0.77		0.59		1.00			

Nota: disponible en formato de diseño de sistemas mecánicos UCCI 2020

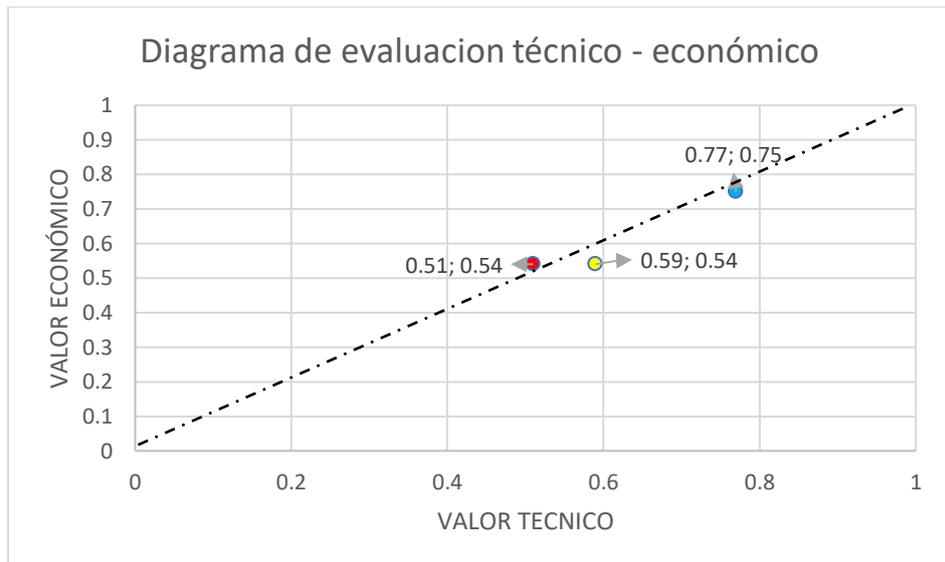
### 4.2.4.2 Valoración económica

**Tabla 13. Valoración económica**

Diseño Mecatrónico		Valoración económica (yi)								Área de diseño	
<b>Proyecto:</b> Aplicativo para la interpretación del Panel Naranja, implementado en un "VANT"											
p: Puntuación de 0 a 4 (Escala de valores según VDI 2206)											
0 = Muy bajo, 1 = Bajo, 2 = Medio, 3 = Regular, 4 = Perfecto (ideal)											
g: Peso ponderado de acuerdo a los criterios de evaluación											
Criterios de evaluación para diseños en fase de proyectos o conceptos											
Variantes de concepto		Solución 1		Solución 2		Solución 3		Solución ideal			
		S1	S2	S3	S Ideal						
N.º	Criterios de calificación	g	p	gp	p	gp	p	gp	p	gp	
1	Costo de programación	6	3	18	3	18	2	12	4	24	
2	Costo de diseño	9	2	18	3	27	2	18	4	36	
3	Costo de mantenimiento	8	2	16	3	24	2	16	4	32	
4	Costo de servidor	6	3	18	3	18	2	12	4	24	
5	Costo de hosting	6	1	6	3	18	3	18	4	24	
Puntaje $\sum gp$		76		105		76		140			
Valor económico yi		0.54		0.75		0.54		1.00			

Nota: disponible en formato de diseño de sistemas mecánicos UCCI 2020

#### 4.2.4.3 Selección de sistema óptimo



**Figura 23. Gráfica de evaluación técnico – económica**

#### Leyenda

Solución 1: color rojo \*

Solución 2: color azul \*

Solución 3: color amarillo \*

Con los datos obtenidos de la evaluación técnico – económica se concluye que la solución óptima es el concepto de solución número 2 (0.77, 0.75), porque se aproxima a la recta de la solución ideal (1,1), como se observa en la figura 24.

#### **La solución 2 obtenida de la matriz morfológica se realizara con el siguiente procedimiento**

Para el desarrollo de la aplicación se hará uso como fuente de alimentación el cargador de una laptop por la portabilidad del sistema, para la visualización de imágenes y la transmisión de video se hará uso de la cámara incorporada en el VANT, para la transmisión de la imagen y video se hará uso del mando del dron, para la recepción de imagen se hará mediante una laptop por su compatibilidad con los sistemas de OCR, para la programación del *back end* se usará Python la api de la aplicación se desarrollará en

HTML, CSS y JS. Además, la base de datos será local mediante archivos de texto, luego para el *front end* se utilizará también Python por su compatibilidad con OCR, el control de aplicativo se hará mediante Windows 10 por la compatibilidad OCR y la facilidad de ejecución de programas en lenguaje Python.

### 4.3 Etapa de diseño

Consiste en desarrollar la fase de diseño de la metodología VDI 2206 adaptada.

#### 4.3.1 Programación de código base

##### 4.3.1.1 Algoritmo de funcionamiento

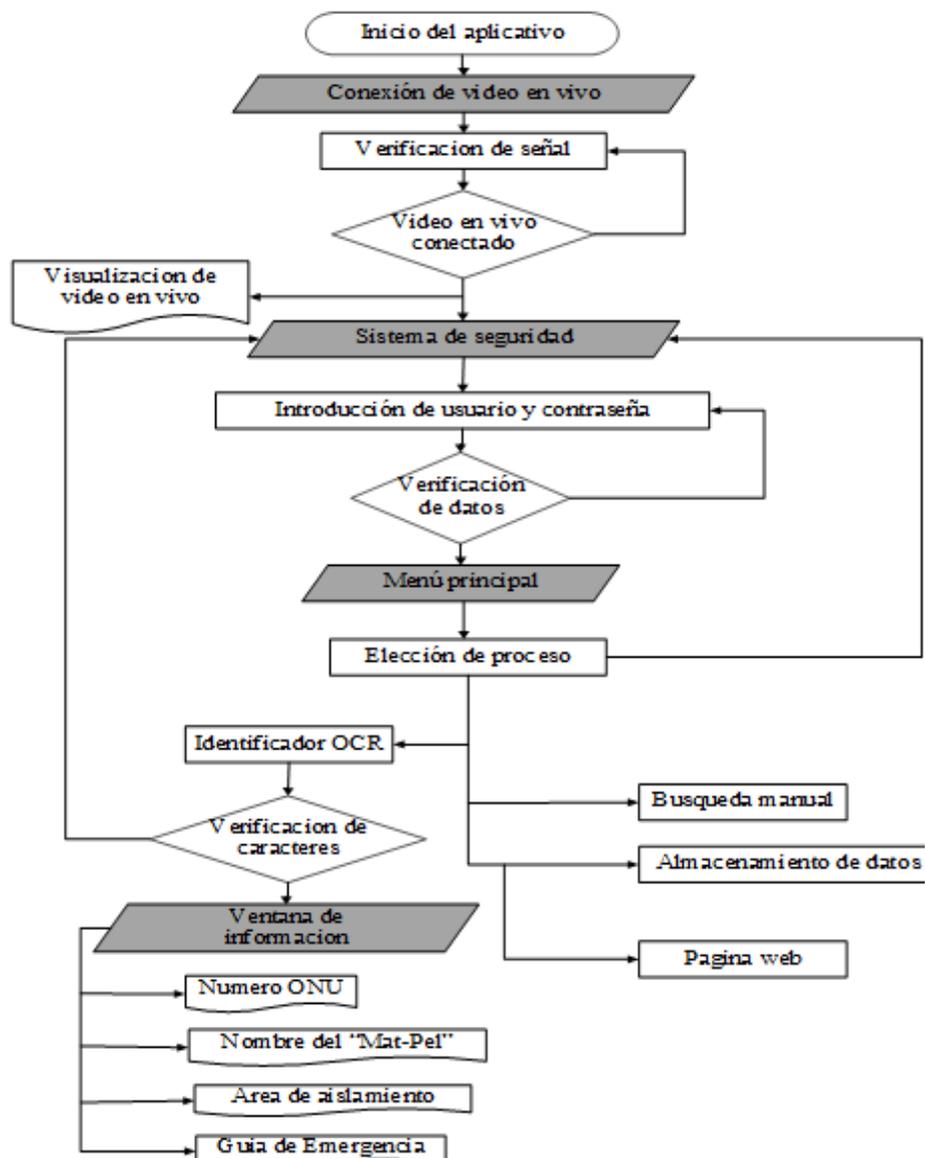


Figura 24. Algoritmo de funcionamiento del aplicativo

#### 4.3.1.2 Código de funcionamiento

Para todo el desarrollo de la aplicación se hará uso del programa *Visual Studio Code*, luego necesitamos instalar Python3 y las librerías “cv2” y “pytesseract”.

Se inicia la programación importando las librerías necesarias:

```
import cv2          #libreria CV2
import pytesseract  #libreria pytesseract
import webbrowser as wb  #libreria webbrowser
import tkinter as tk  #libreria tkinter
```

Luego se definen los parámetros necesarios para el reconocimiento de video.

```
#Vision de video en vivo
cuadro=100
anchocam,altocam=680,400
cap=cv2.VideoCapture(0)
cap.set(3,anchocam)
cap.set(4,altocam)
```

Luego se declara la base de datos.

```
#Files
file='db.txt'
file1='guias.txt'
```

Se definen algunos colores para las botones de las ventanas.

```
#Colores
fondo_entrar="#004AAD" #color azul
fondo_salir="#FF1616" #color azul
fondo_correcto="#8C52FF" #color morado
fondo_incorrecto="#FF5757" #color naranja
fondo_entrada="#D9D9D9" #color humo
botonuso="#FFD700" #color dorado
```

Luego, se crea la ventana principal la cual será el sistema de inicio de sesión o Login.

```
#creacion de ventana principal
ventana = tk.Tk() #creacion de ventana para visualizar los datos
ventana.title("Aplicativo Panel Naranja") #nombre de la ventana
ventana.geometry("500x500+500+50") #tamaño de la ventana
ventana.resizable(width=False,height=False)
ventana.configure(background=fondo_entrada#color de fondo de la ventana
```

Además, se crean los botones y *labels* para la ventana principal.

```
#VENTANA PRINCIPAL
#botones
boton=tk.Button(ventana,text="Entrar",command=login,cursor="hand2",bg=fondo_entrar,width=
12,relief="flat",font=("Comic Sans MS",12,"bold"))
boton.place(x=60,y=405)

boton1=tk.Button(ventana,text="Salir",command=salir,cursor="hand2",bg=fondo_salir,width=12,
relief="flat",font=("Comic Sans MS",12,"bold"))
boton1.place(x=310,y=405)

#entradas
entrada=tk.Entry(ventana,textvar=usuario,width=12,relief="flat",bg="white")
entrada.place(x=230,y=243)

entrada1=tk.Entry(ventana,textvar=password,show="*",width=12,relief="flat",bg="white")
entrada1.place(x=260,y=298)

#label
label=tk.Label(ventana,text="Inicio de sesion",font=("Courier",30),fg="white",bg="black,")
label.place(x=50,y=100)

label1=tk.Label(ventana,text="Usuario",font=("Courier",12),fg="white",bg="black,")
label1.place(x=150,y=240)

label2=tk.Label(ventana,text="Contraseña",font=("Courier",12),fg="white",bg="black,")
label2.place(x=150,y=295)
ventana.mainloop()
```

Luego se definen las variables que almacenarán datos de los cajetines o *labels*.

```
#Variables almacenadas de los cajetines de texto
usuario=tk.StringVar()
password=tk.StringVar()
onu=tk.StringVar()
onualmacenado=tk.StringVar()
nombre=tk.StringVar()
guia=tk.StringVar()
```

se crea la función de identificación de imagen.

```
#funcion de identificador de imagen
def text(image):
    pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract.exe'
    gris=cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
    texto = pytesseract.image_to_string(gris,config='--psm 10 --oem 3 -
c tesseract_ocr_whitelist=UN-0123456789')
    print(texto)
    textox= texto.find("U")
    if textox== -1:
        numeroONUdetectado='No identificado'
        nombre='No identificado'
        guiadeemergencia='No identificado'
        variablex=999
        url='https://identificador-ocr.000webhostapp.com/'

        ventana.withdraw()
        vent=tk.Toplevel()
        vent.title("Error") #nombre de la ventana
        vent.geometry("500x500+500+50") #tamaño de la ventana
        vent.resizable(width=False,height=False)
        vent.configure(background="black") #color de fondo de la ventana
    def salir2():
        ventana.destroy()
        vent.destroy()
        cv2.destroyAllWindows()
    tk.Label(vent,text="Numero ONU DETECTADO",font=("Courier",20),fg="white",bg="black",j
ustify="center").pack(fill=tk.BOTH,expand=True,)
    tk.Label(vent,text=numeroONUdetectado,font=("Courier",20),fg="white",bg="black",justify=
"center").pack(fill=tk.BOTH,expand=True,)
```

```

tk.Label(vent,text=nombre,fg="white",font=("Courier",20),bg="black",justify="center").pack(fill=tk.BOTH,expand=True,)
tk.Label(vent,text=guiadeemergencia,font=("Courier",20),fg="white",bg="black",justify="center").pack(fill=tk.BOTH,expand=True,)
tk.Button(vent,text="Salir",font=("Courier",20),fg="white",bg="black",command=salir2,).pack(fill=tk.BOTH,expand=True,)
vent.mainloop()

print(textox)
texto1 = textox+3;
print(texto1)
texto2=textox+7;
print(texto2)
texto3 = texto[texto1:texto2]
print(texto3)
var=int(texto3)

with open(file) as f_obj:
    lines = f_obj.readlines()

for line in lines:
    str=line.rstrip()
    lista=str.split("|")
    prueba=int(lista[0])
    condicional= prueba==var
    #print(condicional)
    if condicional==True:
        print('encontrado')
        numeroONUdetectado=lista[1]
        nombre=lista[2]
        variablex=lista[3]
        print(lista[1])
        print(lista[2])
        url=lista[4]
        print(lista[4])
        break

    if condicional==False:
        print(condicional)
        numeroONUdetectado='No identificado'

```

```

nombre='No identificado'
variablex=999
url='https://identificador-ocr.000webhostapp.com/'

guiax=int(variablex)
with open(file1) as f_objx:
    linesx = f_objx.readlines()

for line in linesx:
    str=line.rstrip()
    listax=str.split("|")
    pruebax=int(listax[0])
    condicionalx= pruebax==guiax
    #print(condicional)
    if condicionalx==True:
        print('encontrado')
        guiadeemergencia=listax[1]
        print(listax[0])
        print(listax[1])
        pdf=listax[2]
        print(listax[2])
        break

return numeroONUdetectado, nombre, guiadeemergencia,url,pdf

```

Luego, se definen las funciones que cumplirán los botones.

```

#funciones ventanas
def salir(): #funcion para cerrar el programa
    ventana.destroy()

def login(): #funcion de verificacion de datos para acceder al programa
    nombre= usuario.get()
    contraseña=password.get()
    if nombre == "user" and contraseña == "1234": #aqui definimos el usuario y contraseña
        correcta()
    else:
        incorrecta()

```

Si la contraseña no es correcta se activará esta función que dirigirá a otra pantalla, donde permita elegir entre volver a probar la contraseña o cerrar el programa.

```
def incorrecta(): #funcion de contraseña incorrecta
    ventana.withdraw()
    root=tk.Toplevel()          #creacion de ventana para visualizar los datos
    root.title("Error") #nombre de la ventana
    root.geometry("350x350+500+50") #tamaño de la ventana
    root.resizable(width=False,height=False)
    root.configure(background=fondo_entrada) #color de fondo de la ventana

    def regreso():
        root.withdraw()
        ventana.deiconify()

    def salir1():
        root.destroy()
        ventana.destroy()

    #label
    label4=tk.Label(root,text="Contraseña \nincorrecta",font=("Courier",20),fg="white",bg="black",
)
    label4.place(x=80,y=70)

    #botones
    boton4=tk.Button(root,text="Regresar",command=regreso,cursor="hand2",relief="flat",bg=fon
do_correcto,font=("Comic Sans MS",12,"bold"))
    boton4.place(x=120,y=150)

    boton5=tk.Button(root,text="Salir",command=salir1,cursor="hand2",relief="flat",bg=fondo_sali
r,font=("Comic Sans MS",12,"bold"))
    boton5.place(x=140,y=210)

    root.mainloop()
```

Si el usuario y contraseña son correctos se activará la siguiente función que permitirá ingresar al programa y a todas sus funciones.

```

def correcta(): #usuario y contraseña correctos
    ventana.withdraw()
    window = tk.Toplevel() #creacion de ventana de aplicacion
    window.title("Bienvenido") #nombre de la ventana
    window.geometry("500x500+500+50") #tamaño de la ventana
    window.resizable(width=False,height=False)
    window.configure(background=fondo_entrada) #color de fondo

def regreso():
    window.withdraw()
    ventana.deiconify()

def salir2():
    window.destroy()
    ventana.destroy()
    cv2.destroyAllWindows()

def reconocimiento(): #funcion de reconocimiento de caracteres mediante OCR

    while True: #ciclo de captura de video en vivo
        ret,frame=cap.read()
        if ret ==False:break
        cv2.putText(frame,'Ubique el panel naranja',(150,80),cv2.FONT_HERSHEY_SIMPLEX,
0.71,(255,255,0),2)
        cv2.rectangle(frame,(cuadro,cuadro),(anchocam - cuadro, altocam - cuadro),(0,0,0),2)
        x1,y1=cuadro,cuadro
        ancho,alto=(anchocam - cuadro)-x1,(altocam - cuadro) -y1
        x2,y2=x1 +ancho,y1+alto
        doc=frame[y1:y2,x1:x2]
        cv2.imwrite("image.jpg",doc)
        cv2.imshow("Lector Inteligente",frame)
        t=cv2.waitKey(1)
        if t==27:
            cap.release()
            break

def salir2():
    window.destroy()
    ventana.destroy()
    ventanax.destroy()

```

```

cv2.destroyAllWindows()

numeroONUdetectado, nombre, guiadeemergencia,url,pdf=text(doc)
#funcion para redirigir a la pagina web
urlx=url
def link():
    import webbrowser
    global url
    webbrowser.get("C:/Users/User/AppData/Local/Google/Chrome/Application/chrome.exe
%s").open(urlx)

wb.open_new(pdf)
window.withdraw()
ventanax = tk.Toplevel() #ventana donde se mostrara los datos del numero ONU identific
ado
#ventanax = tk.Tk() #creacion de ventana p
ventanax.title("Lector de panel Naranja") #nombre de la ventana
ventanax.geometry("500x500") #tamaño de la ventana
ventanax.configure(background="black") #color de fondo de la ventana
tk.Label(ventanax,text="Numero ONU DETECTADO",font=("Courier",20),fg="white",bg="bl
ack",justify="center").pack(fill=tk.BOTH,expand=True,)
tk.Label(ventanax,text=numeroONUdetectado,font=("Courier",20),fg="white",bg="black",jus
tify="center").pack(fill=tk.BOTH,expand=True,)
tk.Label(ventanax,text=nombre,fg="white",font=("Courier",20),bg="black",justify="center").p
ack(fill=tk.BOTH,expand=True,)
tk.Label(ventanax,text=guiadeemergencia,font=("Courier",20),fg="white",bg="black",justify
="center").pack(fill=tk.BOTH,expand=True,)
tk.Button(ventanax,text="Ver online",font=("Courier",20),fg="white",bg="black",command=li
nk,).pack(fill=tk.BOTH,expand=True,)
tk.Button(ventanax,text="Salir",font=("Courier",20),fg="white",bg="black",command=salir2,)
.pack(fill=tk.BOTH,expand=True,)
ventanax.mainloop()

def manual(): #funcion para ingreso manual de datos
window.withdraw()
rootx=tk.Toplevel() #creacion de ventana para visualizar los datos
rootx.title("Metodo Manual") #nombre de la ventana
rootx.geometry("500x500+500+50") #tamaño de la ventana
rootx.resizable(width=False,height=False)
rootx.configure(background=fondo_entrada) #color de fondo de la ventana

```

```

labelx=tk.Label(rootx,text="Ingrese el numero ONU",font=("Courier",20),fg="white",bg="black",)
labelx.place(x=60,y=100)

#entradas
labelx1=tk.Label(rootx,text="UN-",font=("Courier",14),fg="white",bg="black",)
labelx1.place(x=150,y=200)

entradox1=tk.Entry(rootx,textvar=onu,width=20,relief="flat",bg="white")
entradox1.place(x=190,y=203)

def regresox1():
    rootx.withdraw()
    window.deiconify()

def salir1x():
    rootx.destroy()
    window.destroy()
    ventana.destroy()

def consulta():
    numero= onu.get()
    var=int(numero)

    with open(file) as f_obj:
        lines = f_obj.readlines()

    for line in lines:
        str=line.rstrip()
        lista=str.split("|")
        prueba=int(lista[0])
        condicional= prueba==var
        #print(condicional)
        if condicional==True:
            print('encontrado')
            numeroONUdetectado=lista[1]
            nombre=lista[2]
            variablex=lista[3]
            print(lista[1])

```

```

        print(lista[2])
        url=lista[4]
        print(lista[4])
        break

if condicional==False:
    print(condicional)
    numeroONUdetectado='No identificado'
    nombre='No identificado'
    variablex=999
    url='https://identificador-ocr.000webhostapp.com/'

guiax=int(variablex)
with open(file1) as f_objx:
    linesx = f_objx.readlines()

for line in linesx:
    str=line.rstrip()
    listax=str.split("|")
    pruebax=int(listax[0])
    condicionalx= pruebax==guiax
    #print(condicional)
    if condicionalx==True:
        print('encontrado')
        guiadeemergencia=listax[1]
        print(listax[0])
        print(listax[1])
        pdf=listax[2]
        print(listax[2])
        break
def link():
    import webbrowser
    webbrowser.get("C:/Users/User/AppData/Local/Google/Chrome/Application/chrome.e
xe %s").open(url)
def salir2():
    window.destroy()
    ventana.destroy()
    rootx.destroy()
    ventanax1.destroy()

```

```

wb.open_new(pdf)
rootx.withdraw()
ventanax1 = tk.Toplevel()
#ventanax = tk.Tk() #creacion de ventana para visualizar los datos
ventanax1.title("Lector de panel Naranja") #nombre de la ventana
ventanax1.geometry("500x500") #tamaño de la ventana
ventanax1.configure(background="black") #color de fondo de la ventana
tk.Label(ventanax1,text="Numero ONU DETECTADO",font=("Courier",20),fg="white",bg
="black",justify="center").pack(fill=tk.BOTH,expand=True,)
tk.Label(ventanax1,text=numeroONUdetectado,font=("Courier",20),fg="white",bg="black
",justify="center").pack(fill=tk.BOTH,expand=True,)
tk.Label(ventanax1,text=nombre,fg="white",font=("Courier",20),bg="black",justify="cente
r").pack(fill=tk.BOTH,expand=True,)
tk.Label(ventanax1,text=guiadeemergencia,font=("Courier",20),fg="white",bg="black",jus
tify="center").pack(fill=tk.BOTH,expand=True,)
tk.Button(ventanax1,text="Ver online",font=("Courier",20),fg="white",bg="black",comman
d=link,).pack(fill=tk.BOTH,expand=True,)
tk.Button(ventanax1,text="Salir",font=("Courier",20),fg="white",bg="black",command=sali
r2,).pack(fill=tk.BOTH,expand=True,)
ventanax1.mainloop()

#botones
boton4x=tk.Button(rootx,text="Regresar",command=regreso1,cursor="hand2",relief="flat",
bg=fondo_correcto,font=("Comic Sans MS",12,"bold"))
boton4x.place(x=60,y=400)

boton5x=tk.Button(rootx,text="Salir",command=salir1x,cursor="hand2",relief="flat",bg=fond
o_salir,font=("Comic Sans MS",12,"bold"))
boton5x.place(x=380,y=400)

boton6x=tk.Button(rootx,text="Consultar",command=consulta,cursor="hand2",relief="flat",b
g='#2b5ce3',font=("Comic Sans MS",12,"bold"))
boton6x.place(x=180,y=300)

rootx.mainloop()

#fin de funcion manual

def introducir(): #funcion para aumentar valores a nuestra Base de datos o DB
window.withdraw()

```

```

rootx1=tk.Toplevel() #creacion de ventana para visualizar los datos
rootx1.title("Introducir datos") #nombre de la ventana
rootx1.geometry("500x500+500+50") #tamaño de la ventana
rootx1.resizable(width=False,height=False)
rootx1.configure(background=fondo_entrada) #color de fondo de la ventana

def regresox1():
    rootx1.withdraw()
    window.deiconify()

def salir1x():
    rootx1.destroy()
    window.destroy()
    ventana.destroy()

def almacenar():
    link='https://identificador-ocr.000webhostapp.com/'
    var1= onualmacenado.get()
    var2= nombre.get()
    var3= guia.get()
    final=var1+'|'+UN-'+var1+'|'+var2+'|'+var3+'|'+link
    f = open ('db.txt','a')
    f.write("\n" +final)
    f.close()

labelxxx1=tk.Label(rootx1,text="Ingrese los datos",font=("Courier",20),fg="white",bg="black",)
labelxxx1.place(x=100,y=100)

#entradas
labelx1=tk.Label(rootx1,text="UN-",font=("Courier",14),fg="white",bg="black",)
labelx1.place(x=150,y=170)

entradox2=tk.Entry(rootx1,textvar=onualmacenado,width=20,relief="flat",bg="white")
entradox2.place(x=190,y=173)

labelx2=tk.Label(rootx1,text="Ingrese el nombre: ",font=("Courier",14),fg="white",bg="black",)
labelx2.place(x=20,y=210)

entradox3=tk.Entry(rootx1,textvar=nombre,width=40,relief="flat",bg="white")
entradox3.place(x=240,y=213)

```

```

labelx3=tk.Label(rootx1,text="Guia de emergencia: ",font=("Courier",14),fg="white",bg="black",)
labelx3.place(x=20,y=250)

entradox4=tk.Entry(rootx1,textvar=guia,width=20,relief="flat",bg="white")
entradox4.place(x=250,y=253)

#botones
boton4x=tk.Button(rootx1,text="Regresar",command=regreso1,cursor="hand2",relief="flat",bg=fondo_correcto,font=("Comic Sans MS",12,"bold"))
boton4x.place(x=60,y=400)

boton5x=tk.Button(rootx1,text="Salir",command=salir1x,cursor="hand2",relief="flat",bg=fondo_salir,font=("Comic Sans MS",12,"bold"))
boton5x.place(x=380,y=400)

boton6x=tk.Button(rootx1,text="Almacenar",command=almacenar,cursor="hand2",relief="flat",bg='#2b5ce3',font=("Comic Sans MS",12,"bold"))
boton6x.place(x=180,y=350)

rootx1.mainloop()

#fin de funcion introducir
def pagina(): #funcion para visualizar nuestra pagina web
    import webbrowser
    webbrowser.get("C:/Users/User/AppData/Local/Google/Chrome/Application/chrome.exe %s").open("https://identificador-ocr.000webhostapp.com/")

#botones de ingreso
labelxxx2=tk.Label(window,text="Funciones",font=("Courier",20),fg="white",bg="black",)
labelxxx2.place(x=170,y=30)

boton10=tk.Button(window,text="Cerrar sesion",command=regreso,cursor="hand2",relief="flat",bg=fondo_correcto,font=("Comic Sans MS",12,"bold"))
boton10.place(x=60,y=400)

boton5=tk.Button(window,text="Salir",command=salir2,cursor="hand2",relief="flat",bg=fondo_salir,font=("Comic Sans MS",12,"bold"))
boton5.place(x=380,y=400)

```

```

    boton6=tk.Button(window,text="Identificador OCR",command=reconocimiento,cursor="hand2",relief="flat",bg='#d67b3e',font=("Comic Sans MS",12,"bold"))
    boton6.place(x=170,y=100)

    boton7=tk.Button(window,text="Busqueda manual",command=manual,cursor="hand2",relief="flat",bg='#d67b3e',font=("Comic Sans MS",12,"bold"))
    boton7.place(x=175,y=170)

    boton7=tk.Button(window,text="Almacenamiento de datos",command=introducir,cursor="hand2",relief="flat",bg='#d67b3e',font=("Comic Sans MS",12,"bold"))
    boton7.place(x=150,y=240)

    botonxx7=tk.Button(window,text="Pagina Web",command=pagina,cursor="hand2",relief="flat",bg='#d67b3e',font=("Comic Sans MS",12,"bold"))
    botonxx7.place(x=190,y=310)

    window.mainloop()

#fin del programa

```

### 4.3.2 Programación de código web

La programación de la página web se realizará usando la interacción de tres programas (HTML5, CSS, Java Script), para luego *hoostearla* en un servidor en línea para la visualización del personal de bomberos y de las personas que deseen saber la información.

#### 4.3.2.1 Código de página principal

##### Código de HTML5

Se inicia la página web con la etiqueta head el cual se encarga de brindar información global del documento (metadatos), como lo es el título, enlaces de scripts y por último dirige al código CSS.

```

<!--Inicialización de la Pagina web-->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">

```

```

<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>App Panel Naranja</title>
<!-- FONT AWESOME: para poder insertar iconos -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
<!--FONT STIX:para poder insertar tipo de letras-->
<link href="https://fonts.googleapis.com/css2?family=STIX+Two+Text:ital,wght@0,400;0,500;
1,400&display=swap" rel="stylesheet">
<!--Personalizacion con CSS -->
<link rel="stylesheet" href="styles.css">
</head>

```

Se sigue estructurando el cuerpo de la página web, todo este proceso se realiza en la etiqueta body, aquí se desarrollarán todos elementos que contendrá la página web.

Aquí se muestra el desarrollo del título de la aplicación, la barra de navegación tanto en su visualización desde un PC y un celular.

```

<!--Título de la Aplicación-->
<H1 class="titulo">APLICATIVO PANEL NARANJA</H1>
<!--Barra de navegacion de Numeros ONU Cuando se vea por una PC-->
<div class="menu-btn">
  <i class="fas fa-bars fa-2x"></i>
</div>
<div class="container">
  <nav class="nav-main">
    
    <ul class="nav-menu show">
      <i>
        <a href="panel/index1006.html">UN-1006</a>
      </li>
      <i>
        <a href="panel/index1013.html">UN-1013</a>
      </li>
      <i>
        <a href="panel/index1049.html">UN-1049</a>
      </li>
    </ul>
  </nav>

```

```

        </li>
        <a href="#">More</a>
    </li>
</ul>
<!--Barra de navegación de Números ONU cuando se vea por celular-->
<ul class="nav-menu-right">
    <li>
        <a href="#">
            <i class="fas fa-search"></i>
        </a>
    </li>
</ul>

</nav>
<hr>

```

En esta sección se verá el desarrollo de la portada principal, la portada secundaria y las tarjetas de información, estos tendrán la información necesaria sobre la investigación realizada, como también se informará sobre las últimas actualizaciones que tiene la guía GRE2020.

```

<!--Portada Principal-->
<header class="showcase">
    <h2>Big Nwes Today</h2>
    <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Minus qui alias sunt tempori bus. Blanditiis libero, harum nostrum ut esse, dolorum vitae, quidem iure nihil voluptas maxime explicabo laudantium ipsum laborum?</p>
    <a href="#" class="btn">Read More <i class="fas fa-angle-double-right"></i></a>
</header>
<!--Tarjetas de Informacion-->
<div class="news-cards">
    <div>
        
        <h3>Lorem ipsum dolor </h3>
        <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Perspiciatis illum dicta eius velit soluta quaerat veniam quo? Quas neque consequuntur, optio suscipit nam nemo nihil in, rerum, minima error harum.</p>
        <a href="#">Leer Mas <i class="fas fa-angle-double-right"></i></a>
    </div>

```

```

<div>
  
  <h3>Lorem ipsum dolor </h3>
  <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Perspiciatis illum dicta eius velit soluta quaerat veniam quo? Quas neque consequuntur, optio suscipit nam nemo nihil in, rerum, minima error harum.</p>
  <a href="#">Leer Mas <i class="fas fa-angle-double-right"></i></a>
</div>
<div>
</div>

<!--Portada Secundaria-->
<section class="cards-banner-one">
  <div class="content">
    <h2>Lorem, ipsum dolor.</h2>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Corrupti amet molestias fugiat cumque. Nisi nesciunt ex, harum ea id asperiores.</p>
    <a href="#" class="btn">Leer Mas <i class="fas fa-angle-double-right"></i></a>
  </div>
</section>

```

La página web también tendrá una sección de redes sociales y un pie de página, estas secciones están estandarizadas y se tienen que mostrar en todos los sitios web por cuestión de estética y profesionalismo.

```

<!--Links de las redes Sociales-->
<section class="social">
  <p>Follow </p>
  <div class="links">
    <a href="#">
      <i class="fab fa-facebook-f"></i>
    </a>
    <a href="#">
      <i class="fab fa-twitter"></i>
    </a>
    <a href="#">
      <i class="fab fa-linkedin"></i>
    </a>
  </div>

```

```

</section>
</div>
<!--Pie de pagina de la web-->
<div class="footer-links">
  <div class="footer-container">
    <ul>
      <li>
        <a href="#">
          <h3>Title one</h3>
        </a>
      </li>
      <li>
        <a href="#">Reviews</a>
      </li>
    </ul>
  </div>
</div>

```

Por último, pero no menos importante, se tendrá que desarrollar la parte del copyright esta sección es necesaria para poder tener derechos sobre nuestra página web creada, también está la sección donde se detallan los scripts de java script (no aparece en la página web) este permite poder llamar las configuraciones que se realiza en dicho programa, para esta página

web se desarrolló una función que va mostrando la información mientras se va desplazando con el *scroll* del mouse.

```
<!--Derechos de Copyright-->
<footer class="footer">
  <h3>Reconocimiento @Copyright</h3>
</footer>

<!--Animacion de Java Script-->
<script src="https://unpkg.com/scrollreveal"></script>
<!--Personalizacion Js-->
<script src="main.js"></script>

</body>
```

### Código de CSS

El programa CSS se encarga de estilizar todas las funciones que tienen el programa HTML5.

Primero se le da un estilo apropiado a la página principal, el cuerpo de la página, el título de la aplicación y el encabezamiento de la aplicación. Todo esto se realiza primero para poder tener la página web lista para poder darle estilo al resto de las etiquetas de HTML5.

```
/* Estilo de la página principal*/
*{
  box-sizing: border-box;
  margin: 0;
  padding: 0;
}
/*Estilo del cuerpo de la pagina*/
body{
  background-color: #000000;
  font-family: 'STIX Two Text', serif;
  color: #fff;
}
/*Estilo del titulo de la Aplicacion*/
.titulo{
  text-align: center;
```

```
padding-top: 6px;
margin: auto;
font-size: 25px;
color: aqua;
}
/*Estilo del encabezamiento de la aplicacion*/
a{
color: #fff;
text-decoration: none;
}
ul {
list-style: none;
}
.container {
width: 90%;
margin: auto;
}
}
```

Ahora, se le da estilo a la barra de navegación de los números ONU, esta barra tiene que poder visualizar desde una pantalla de PC, una Tablet y un celular sin ningún problema, para esto es necesario darle un formato correcto en CSS.

```
/* Estilo de la barra de navegacion de Numeros ONU cuando se vea por una PC*/
.nav-main {
font-size: 17px;
display: flex;
justify-content: space-between;
align-items: center;
height: 60px;
padding: 20px 0;
}
.nav-brand {
width: 50px;
}
/* Estilo de la barra de navegacion de Numeros ONU cuando se vea por un Celular*/
.nav-main ul {
display: flex;
}
.nav-main ul li {
```

```

padding: 10px;
}
.nav-main ul li a{
padding: 6px;
}
.nav-main ul li a:hover {
border-bottom: 2px solid #158256;
background: linear-gradient(to bottom,transparent,#158256);
}

.nav-main ul.nav-menu {
flex: 1;
margin-left: 20px;
}

.menu-btn {
position: absolute;
display: none;
cursor: pointer;
top: 15px;
right: 30px;
z-index: 2;
}

.menu-btn:hover {
background: linear-gradient(to bottom,transparent,#158256);
}

```

Luego, se tendrá que dar un estilo a la portada principal, tarjetas de noticias, y la portada principal, esto se tiene que realizar manteniendo el orden de la página principal.

```

/*Estilo de la portada Principal*/
hr {
margin: 10px 0;
}
.showcase {
width: 100%;
height: 550px;
background: url(img/showcase1.jpeg) no-repeat center center/cover;
}

```

```

padding: 50px;
margin-bottom: 20px;
display: flex;
flex-direction: column;
align-items: center;
justify-content: flex-end;
}
.btn {
  cursor: pointer;
  display: inline-block;
  border: 0;
  font-weight: bold;
  padding: 10px 20px;
  background: #3B33DF;
  color: #111;
  font-size: 15px;
  border: 2px solid #1E17B2;
  margin: 10px 0;
}
.btn:hover {
  opacity: .7;
}
.showcase h2,
.showcase p {
  margin-bottom: 10px;
}
/*Estilo de la tarjetas de Informacion*/
.news-cards {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr 1fr;
  gap: 25px;
  margin: 10px 0;
}
.news-cards img {
  width: 100%;
  height: 180px;
}
/*Estilo de la portada Secundaria*/
.cards-banner-one {
  width: 100%;

```

```

height: 350px;
background: url(/img/showcase2.jpeg)no-repeat center center/cover;
margin-bottom: 40px;
}
.cards-banner-one .content {
width: 40%;
padding: 90px 0 0 30px;
color: #fff;
}
.cards-banner-one p,
.cards-banner-one h2 {
margin: 10px 0 20px 0;
}

```

Ahora, se le da un estilo a la parte de las redes sociales, el pie de página y los derechos del copyright.

```

/*Estilo de los links de las redes sociales*/
.social {
margin: 50px;
}
.social p {
text-align: center;
font-size: 30px;
margin-bottom: 20px;
}
/*Estilo del pie de pagina de la Web*/
.footer-links {
background: #2f3640;
color: #616161;
font-size: 12px;
padding: 35px 0;
}
.footer-container {
display: grid;
grid-template-columns: repeat(4, 1fr);
gap: 10px;
align-items: flex-start;
justify-content: center;
}
.footer-container ul {

```

```

margin: 0 auto;
}
.footer-container ul li {
  line-height: 2.6;
}
/*Estilo de Derechos de Copyright*/
.footer {
  background: #2f3640;
  color: #616161;
  font-size: 12px;
  padding: 20px 0;
  text-align: center;
  padding: 20px;
}

```

Tal y como se hizo con la barra de navegación para que se pueda ver en los distintos dispositivos (PC, Tablet, celular) también se tiene que dar un formato a toda la página web, para así no tener problemas al momento de visualizarlo desde cualquier dispositivo.

```

/*Estilo para ver la pagina web desde una Tablet*/

```

```

@media (max-width: 700px) {
  .menu-btn{
    display: block;
  }
  .nav-main ul.nav-menu{
    display: block;
    position: absolute;
    top: 0;
    left: 0;
    background: #3A3C46;
    height: 100%;
    padding: 20px;
    opacity: .8;
    transform: translateX(-400px);
    transition: transform ,2s ease-in-out;
  }
  .nav-main ul.nav-menu.show {
    transform: translateX(-20px);
  }
}

```

```

}
.nav-main ul.nav-menu li {
  padding: 20px;
  font-size: 14px;
}
.nav-main ul.nav-menu-right {
  padding-top: 5px;
  font-size: 1.4rem;
  margin-right: 40px;
}
}
/*Estilo para ver la pagina web desde un celular*/
@media (max-width: 500px) {
  .news-cards {
    grid-template-columns: 1fr;
  }
  .news-cards img{
    height: 270px;
  }
  .cards-banner-one .content,
  .cards-banner-two .content{
    width: 100%;
    padding: 60px 20px;
  }
  .footer-links .footer-container{
    grid-template-columns: 1fr;
  }
}
}

```

### Código de Java Script

Para esta página web solo se usaron dos configuraciones, el primero es para la barra de navegación el cual permite que cuando se vea desde una Tablet o un celular la barra de navegación se visualice haciendo un clic. La segunda configuración es para ver la información de la página web mediante el *scroll* del mouse, al momento de bajar la información irá apareciendo de manera gradual.

```
//Programacion para que la barra de navegacion se ejecute con un click cuando este en modo c
elular y tablet
```

```

document.querySelector('.menu-btn').addEventListener('click', () =>{
  document.querySelector('.nav-menu').classList.toggle('show')
})
//Programacion para que la infomacion de la pagina web aparezca mediante desplaces el scrol
|
ScrollReveal().reveal('.showcase');
ScrollReveal().reveal('.news-cards', {delay:500});
ScrollReveal().reveal('.cards-banner-one', {delay:500});
ScrollReveal().reveal('.cards-banner-two', {delay:500});

```

### 4.3.2.2 Código de la página del panel de información

#### Código HTML5

Se inicia la página web del panel de información de la misma manera que la página web principal, esto es necesario, ya que las páginas necesitan de esta inicialización para poder funcionar.

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Panel Naranja</title>
  <!-- FONT AWESOME:para poder insertar iconos -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
  <!--FONT STIX:para poder insertar tipo de letras-->
  <link href="https://fonts.googleapis.com/css2?family=STIX+Two+Text:ital,wght@0,400;0,500;
1,400&display=swap" rel="stylesheet">
  <link rel="stylesheet" tybe="text/css" href="estilo.css">
  <link rel="stylesheet" tybe="text/css" href="normalize.css">
  <link rel="stylesheet" href="https://bootswatch.com/5/materia/bootstrap.min.css">
  <link href="https://fonts.googleapis.com/css2?family=Gowun+Dodum&display=swap" rel="sty
lesheet">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script src="https://kit.fontawesome.com/bafc4298b2.js" crossorigin="anonymous"></script>
</head>

```

Después se hace una pequeña barra de navegación, esto para poder comunicarse con la página web principal, y al momento de darle clic, se pueda regresar a la página principal.

```
<body class="body">
  <!--Barra de navegacion Para regresar a la pagina Principal-->
  <div class="menu-btn">
    <i class="fas fa-bars fa-2x"></i>
  </div>
  <div class="container">
    <nav class="nav-main">
      
      <ul class="nav-menu-show" >
        <li>
          <a href="../index.html" class="a">APLICATIVO PANEL NARANJA</a>
        </li>
      </ul>
      <ul class="nav-menu-right">
        <li>
          <a class="a" href="#">
            <i class="fas fa-search"></i>
          </a>
        </li>
      </ul>
    </nav>
  <hr>
```

Y por último, se realiza el panel, el cual se encargará de mostrar la información necesaria que se obtenga de la aplicación.

```
<!--Panel para mostrar la informacion adquirida de la aplicacion-->
  <div class="flex-container">
    <form class="form">
      <div class="form__section">
        <input type="text" class="form__input" placeholder="Numero ONU : UN - 1006 ">
      </div>
      <div class="form__section">
        <input type="email" class="form__input" placeholder="Nombre del 'Mat-Pel' : Argon">
      </div>
      <div class="form__section">
        <input class="form__input" placeholder="Guia de emergencia 120 "></input>
      </div>
    </form>
  </div>
```

```

</div>
<div class="d-grid gap-2 pt-4">
  <button type="submit" class="btn btn-success ">
    <a href="https://drive.google.com/file/d/15myoAfHtDxXj3JTfk3LCuOXF9MfyYX-
B/view" target="_BLANK" class="btn btn-success ">VER GUIA</a>
  </button>
</div>
</form>
<!--Imagen Principal de la guia Gree2020-->
<div class="form-img">
  
</div>
</div>
</body>
</html>

```

## Código CSS

Las configuraciones son las mismas que las realizadas en la página principal, el único cambio significativo es el del panel de información.

```

/*Estilo Panel para mostrar la informacion adquirida de la aplicacion*/
.flex-container{
  display: flex;
  flex-wrap: wrap;
  align-items: center;
  justify-content: center;
  background: #333;
  padding: 40px;
  margin: 50px 50px;
  border-radius: 30px;
}
.form__section input::placeholder{
  color: #BBE1CF;
}
.form__section textarea::placeholder{
  color: #BBE1CF;
}
.form{

```

```

min-width: 300px;
flex: 1;
text-align: center;
padding: 30px;
order: 5;
}
.form__input{
width: 100%;
padding: 7px;
margin: 6px 0;
border: none;
border-bottom: 4px solid #23F385;
background: transparent;
color: #fff;
}
.form__input:focus{
background: linear-gradient(to bottom,transparent,#222);
outline: none;
border-bottom: 4px solid #5BEFA8;
}

.form textarea{
min-height: 150px;
resize: none;
}
.form-img{
flex: 1;
display: flex;
min-width: 300px;
max-width: 400px;
order: 2;
}
.form-img img{
width: 80%;
margin: auto;
background: #444;
}
.form__input[type="submit"]{
background-color: #229D63;
border-bottom: none;

```

```
padding: 15px;
}
.form__input[type="submit"]:hover{
background-color: #229D63;
border-bottom: none;
padding: 15px;
}
.form__input[type="submit"]:focus{
background: #072;
}
.form__input[type="submit"]:active{
background: linear-gradient(to bottom,#229D63,#222);
}
```

## CAPÍTULO V CONSTRUCCIÓN

### 5.1 Construcción

Consiste en desarrollar la fase de codificación de la metodología VDI 2206 adaptada.

#### 5.1.1 Inicio del aplicativo

Para ingresar al aplicativo primero se debe ingresar un usuario y contraseña como se puede visualizar en la figura 25.



*Figura 25. Inicio de sesión del aplicativo*

Si el usuario y contraseña son incorrectos aparecerá una nueva ventana donde da la posibilidad de intentar nuevamente o salir del aplicativo como se puede visualizar en la figura 26.



**Figura 26. Ventana de error de datos**

Si el usuario y contraseña son correctos aparecerá la siguiente ventana donde se mostrarán todas las funciones del aplicativo como se puede visualizar en la figura 27.



**Figura 27. Menú de funciones del aplicativo**



**Figura 28. Lector inteligente**

### 5.1.2 Identificador OCR

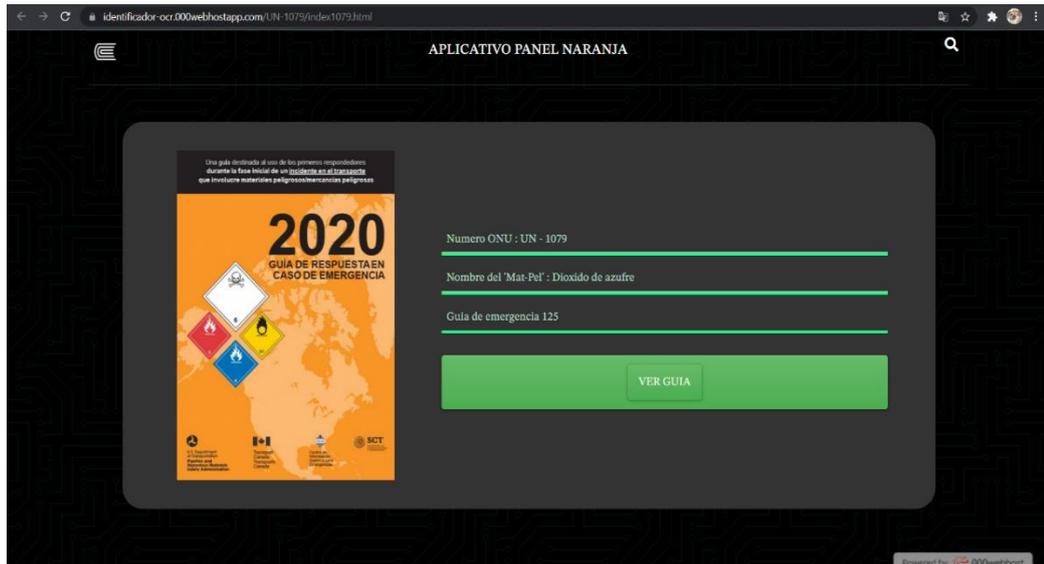
La primera función se llama Identificador OCR el cual al acceder se abrirá una ventana donde se visualizará la cámara que detectará el Panel Naranja, como se muestra en la figura 28.

Esta ventana tiene un cuadrado indicador que sirve para posicionar el Panel Naranja y una vez posicionado se presiona la tecla “ESC” para identificar el número ONU capturado y entonces se obtiene de resultado el nombre del “Mat-Pel”, número ONU y la guía de emergencia que le corresponde como se puede ver en la figura 29.

```
Numero ONU DETECTADO: UN-1075
Nombre: Gases licuados de petroleo(GLP)
  ↓ Recomendaciones ↓
  Aislar el area del derrame o
  escape de Mat-Pel en todas las direcciones
  Distancia de aislamiento:
  Minimo 100 metros
Prohibido acercarse si no cuenta con sus EPP
Intervenir a favor del viento
Consultar Guia de emergencia 115 en la GREE 2020 para mas informacion
Abrir Guia de emergencia 115
Ver online
Salir
```

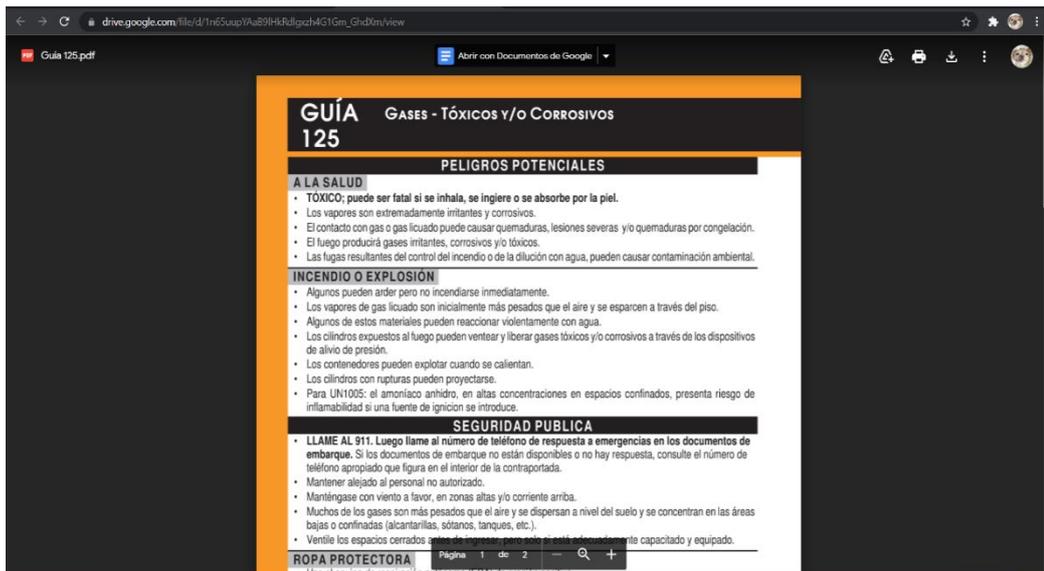
**Figura 29. Número ONU identificado**

Además, se tiene la opción de “Ver online” que abrirá la página web con el número ONU detectado como se puede visualizar en la figura 30.



**Figura 30. Aplicativo online**

Además, si se da click al botón que dice “ver guía” se abrirá la Guía de emergencia de forma virtual como se puede visualizar en la figura 31.



**Figura 31. Guía de emergencia online**

### 5.1.3 Búsqueda manual

Esta función permite obtener el nombre del “Mat-Pel” y la guía de emergencia ingresando el número ONU de manera manual como se puede apreciar en la figura 32.

Además, incluye un botón de regreso vinculado al menú principal y un botón de salida para cerrar el aplicativo.

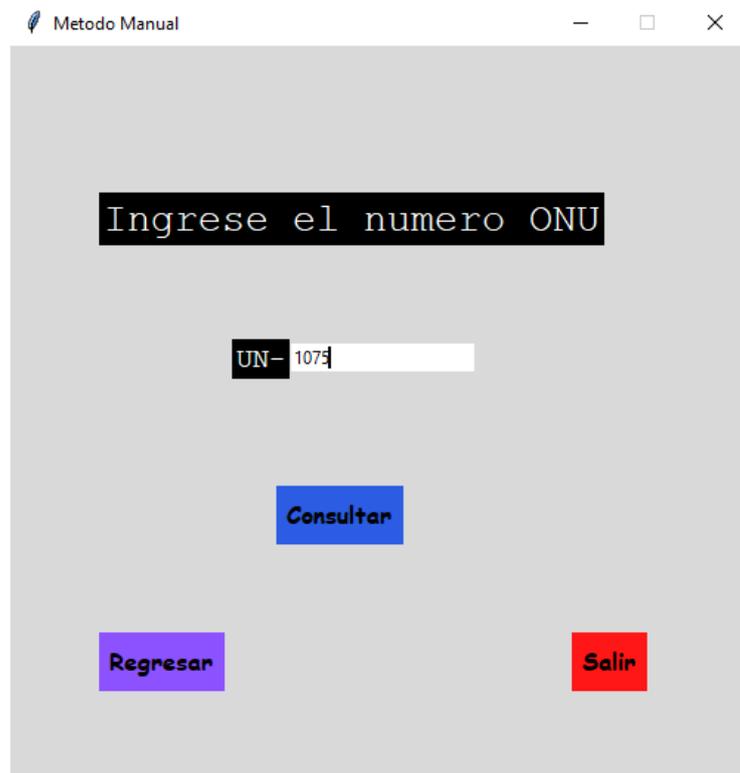


Figura 32. Función de búsqueda manual

Si el número ONU ingresado se encuentra en la base de datos, se mostrará una ventana con el nombre del “Mat-Pel”, la guía de emergencia en formato PDF y un enlace a la página web, si se quiere obtener la guía de forma virtual como se puede visualizar en la figura 33.

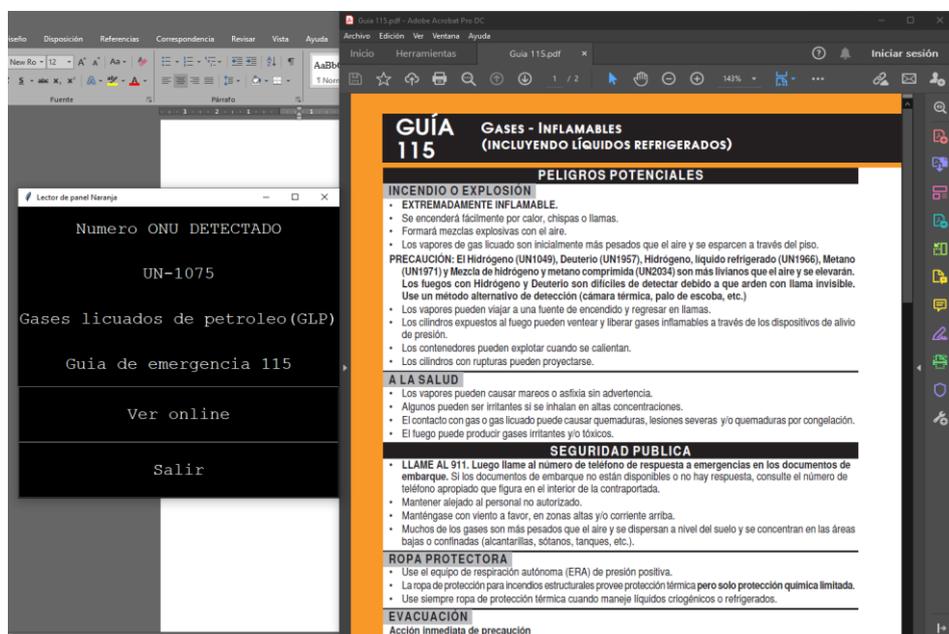


Figura 33. Resultados de búsqueda manual

### 5.1.4 Almacenamiento de datos

Esta función permite incrementar la base de datos de los números ONU, ingresando el número ONU, el nombre del “Mat-Pel” y la guía de emergencia como se observa en la figura 34, todos estos datos deben ser verificados en la GREE 2020.

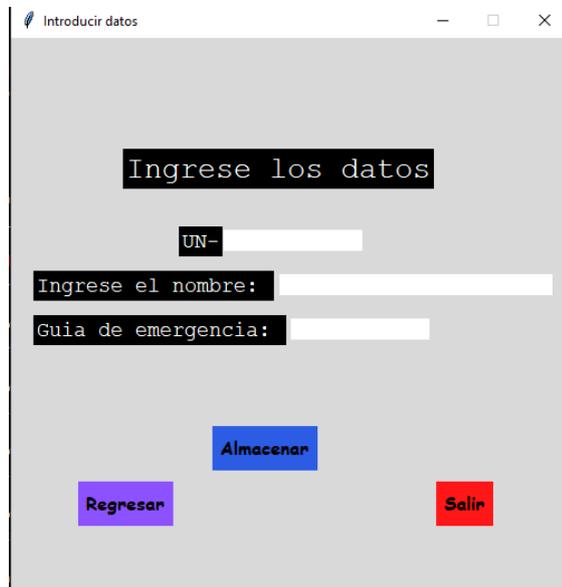


Figura 34. Función de almacenamiento de datos

### 5.1.5 Pagina web

Esta función abre el navegador para visualizar la página web del aplicativo donde se pueden encontrar los números ONU más comunes en Perú y todas las guías de emergencia en formato virtual como se puede visualizar en las figuras 35 y 36.

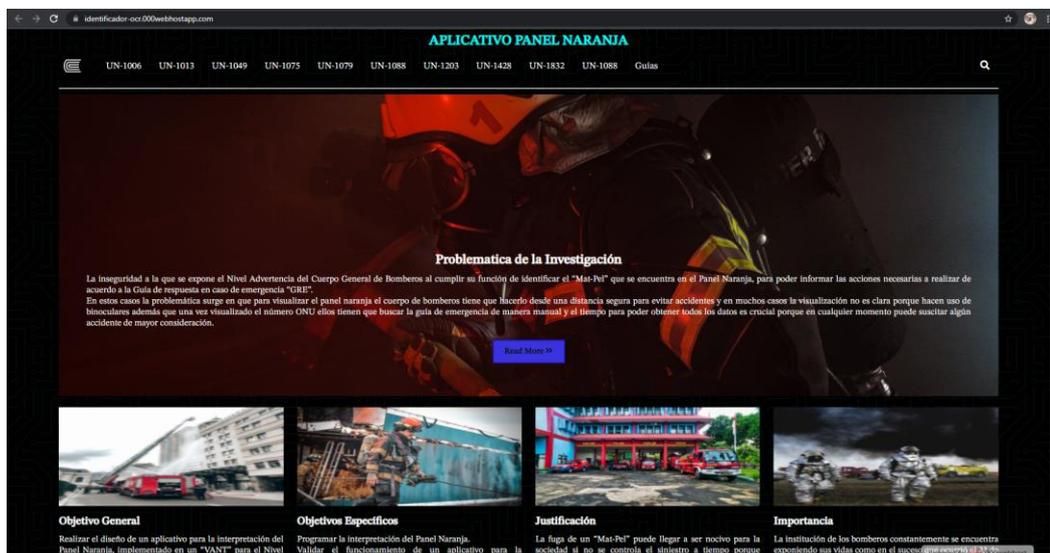


Figura 35. Página web del aplicativo

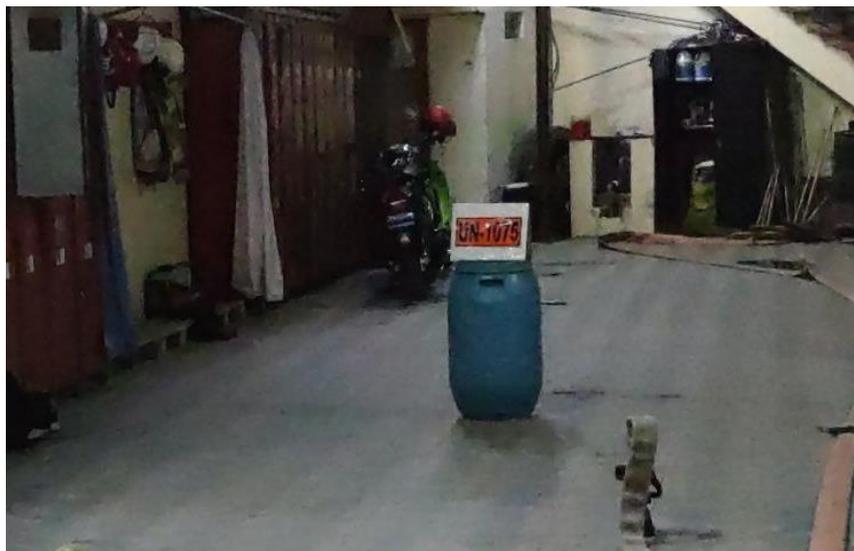
GUIA DE EMERGENCIA	OPERACION
Guia de Emergencia 111	VER
Guia de Emergencia 112	VER
Guia de Emergencia 113	VER
Guia de Emergencia 114	VER
Guia de Emergencia 115	VER
Guia de Emergencia 116	VER
Guia de Emergencia 117	VER
Guia de Emergencia 118	VER
Guia de Emergencia 119	VER
Guia de Emergencia 120	VER
Guia de Emergencia 122	VER
Guia de Emergencia 123	VER
Guia de Emergencia 124	VER
Guia de Emergencia 125	VER
Guia de Emergencia 126	VER
Guia de Emergencia 127	VER
Guia de Emergencia 128	VER

**Figura 36. Guías de emergencia en la página web**

## 5.2 Pruebas y resultados

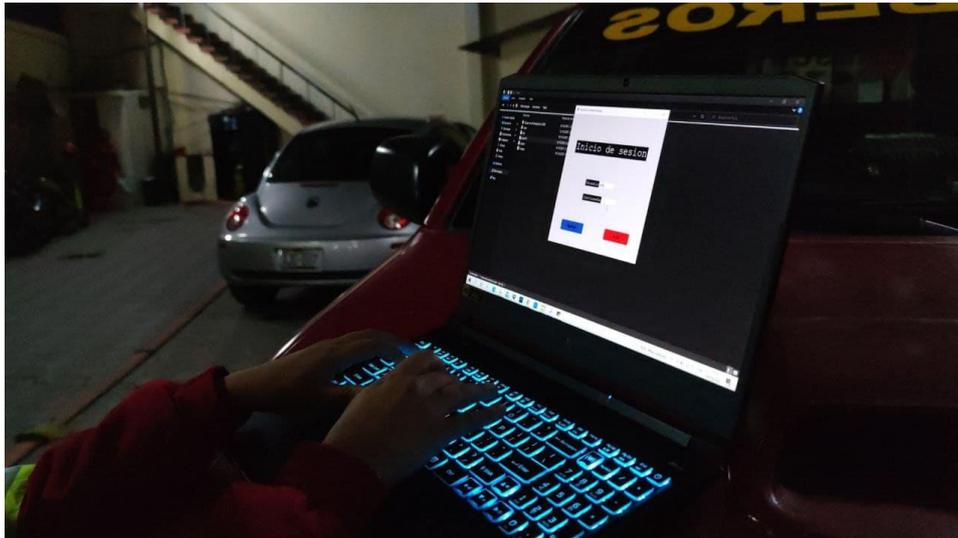
Las pruebas se realizaron con apoyo de la compañía de bomberos en 2 oportunidades de acuerdo a la disponibilidad de su tiempo.

Para la primera prueba se hizo la escenificación de una fuga de un “Mat-Pel” que se encontraba en un contenedor como se puede visualizar en la figura 37.



**Figura 37. Fuga de un contenedor**

Entonces, de acuerdo al protocolo de intervención, el nivel Advertencia tiene que identificar el “Mat-Pel” del contenedor para poder ejecutar las diligencias respectivas, entonces ingresaron al aplicativo introduciendo el usuario y contraseña como se puede visualizar en la figura 38.



**Figura 38. Inicio de sesión del aplicativo**

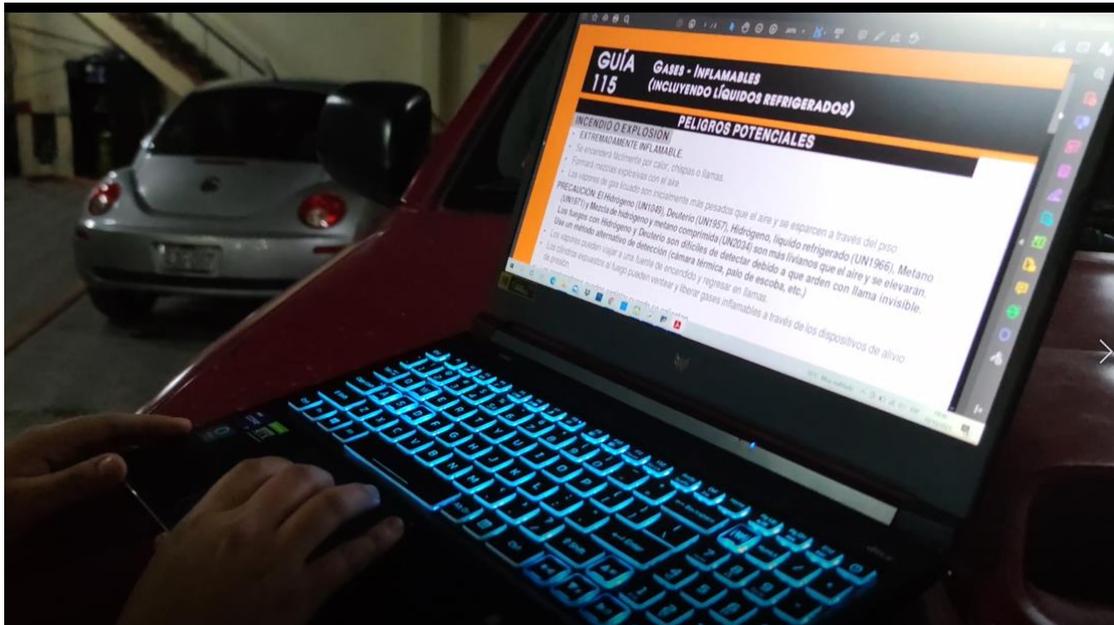
Si el usuario y contraseña son correctos se accederá al menú del aplicativo donde se encuentra el Identificador OCR como se puede visualizar en la figura 39.



**Figura 39. Identificador OCR**

En la aplicación se capturó el Panel Naranja para su posterior identificación como se observa en la figura 40.





**Figura 42. Guía de emergencia asignada al “Mat-Pel”**

Con la información obtenida de la guía de emergencia se procede a comunicar al personal todos los datos necesarios para una posterior intervención como se puede visualizar en las figuras 43 y 44.



**Figura 43. Comunicación de datos al personal de bomberos**



**Figura 44. Intervención al área del siniestro**

Para la segunda prueba se recrea una fuga de “Mat-Pel” que transportaba un vehículo como se puede visualizar en la figura 45.



**Figura 45. Fuga de “Mat-Pel” transportado en un vehículo**

De igual forma que en la primera prueba se requiere el inicio de sesión con los datos correctos para acceder al aplicativo como se puede visualizar en la figura 46.



**Figura 46. Inicio de sesión prueba 2**

Una vez iniciada la sesión el drone se tiene que posicionar correctamente de manera que se pueda reconocer el Panel Naranja mediante el aplicativo como se puede observar en las figuras 47 y 48.

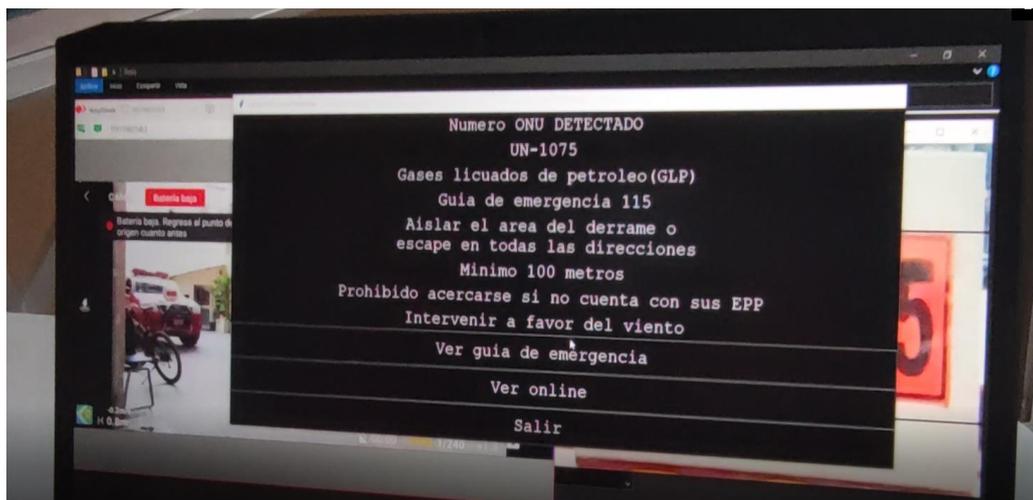


**Figura 47. Despegue del drone**



**Figura 48. Posicionamiento del dron**

Una vez posicionado el dron se procede a capturar la imagen del video en vivo emitido por el dron para su procesamiento en el aplicativo y finalmente mostrar los datos después de identificar el número ONU como se puede visualizar en la figura 49.



**Figura 49. Datos extraídos del aplicativo**

Obtenidos los datos del “Mat-Pel”, los cuales son el nombre Gases Licuados de Petróleo (GLP) , la guía de emergencia 115 que contiene todos los datos de intervención de acuerdo al “Mat-Pel”, el área de aislamiento que es de 100 metros para este “Mat-Pel” y los consejos de intervención como es la prohibición de acercarse al área del incidente si no cuenta con los EPP

adecuadas y trabajar a favor del viento para evitar el contacto por la partículas suspendidas en el aire. Con estos datos el personal de bomberos puede acercarse a dar una revisión del incidente como se puede visualizar en la figura 50.



***Figura 50. Intervención del personal de bomberos con los datos obtenidos del aplicativo***

## CONCLUSIONES

- Se logró diseñar un aplicativo para la interpretación del Panel Naranja, implementado en un “VANT” que permitió optimizar el tiempo de identificación del Panel Naranja, obteniendo los datos del “Mat-Pel” desde una distancia segura salvaguardando la integridad del nivel Advertencia del Cuerpo General de Bomberos de la región Junín.
- La interpretación del Panel Naranja se programó en el lenguaje de programación Python por su compatibilidad con el reconocimiento óptico de caracteres (OCR), que permitió extraer el texto a través de una imagen capturada del video en vivo emitido de la cámara del dron, para su posterior identificación mediante la comparación del texto extraído con la base de datos, del cual se obtuvo el nombre del “Mat-Pel” , la guía de emergencia asignada, consejos de intervención y la distancia de aislamiento para una evacuación rápida.
- La validación del funcionamiento se realizó en 2 pruebas con el apoyo del Capitán CBP Aryf Torres Orihuela García y la compañía de bomberos Cesar A. Habich Sosa N.º 30 en la cual escenificamos una fuga de “Mat-Pel” de un contenedor abandonado y la fuga de “Mat-Pel” de un vehículo que tuvo un accidente en ambos casos el “Mat-Pel” fue el UN-1075 y mediante la aplicación se obtuvo el nombre del “Mat-Pel”, el cual es Gases Licuados de Petróleo (GLP) junto con la guía de emergencia 115 donde se encuentra los peligros potenciales, la seguridad pública y la respuesta ante emergencia de una fuga o derrame de GLP. Además, el área de aislamiento que sería 100 metros en todas las direcciones y los consejos útiles como evitar acercarse si el personal de bomberos no cuenta con las EPP adecuadas y si se interviene trabajar a favor del viento para evitar contacto con el material mientras se resuelve el siniestro.

## TRABAJOS FUTUROS

- Se recomienda adaptar el aplicativo para que pueda utilizarse en multiplataformas.
- Se recomienda incrementar la información de la base de datos de acuerdo al uso que necesite.
- Se recomienda tener una buena conectividad de internet para la transmisión de video del dron hacia el aplicativo.
- Dependiendo del modelo del dron se puede incrementar el área de alcance del aplicativo.

## LISTA DE REFERENCIAS

1. **LAZO, Noé.** *Primera respuesta a incidentes con materiales peligrosos Primap.* Lima : s.n., 2019. págs. 27, 28.
2. **RPP.** *Al menos dos muertos y 48 heridos tras deflagración por fuga de gas de un camión cisterna en Villa El Salvador.* 2020, Diario Correo.
3. **KHARUF GUTIÉRREZ, Samy, y otros.** *Análisis de imágenes multiespectrales adquiridas con vehículos aéreos no tripulados.* Universidad Central "Marta Abreu", Santa Clara, Cuba : 2018.
4. **CAMPOS, S. Madridano; GARCÍA, F. Al-Kaff; ESCALERA, A. Martín.** *Vehículo aéreo no tripulado para vigilancia y monitorización de incendios.* Universidad Carlos III de Madrid, Madrid, España : 2020.
5. **HELGUERO VELÁSQUEZ, Hernán; MEDINACELI TÓRREZ, Rubén.** *Robot minero: sistema detector de gases utilizando sensores en tiempo real MIN – SIS 1.0 SDG-STR.* Scielo.org, Oruro, Bolivia : 2019.
6. **POSTIGO HUANQUI, Sergio Renato.** *Vehículo aéreo no tripulado para vigilancia en ambientes cerrados con detección de personas y obstáculos a su alrededor.* Pontificia Universidad Católica del Perú, Lima : 2018.
7. **MORZAN SOTO, Arturo Alejandro; ZERGA TAPIA, Bruno Giuseppe.** *Diseño e implementación de robot terrestre teleoperado aplicando imágenes térmicas para mitigar incendios en residenciales operado por bomberos Victoria N.º 8.* Universidad Ricardo Palma, Lima-Perú : 2018.
8. **MUÑOZ MARTÍNEZ, César James.** *Diseño e implementación de un robot móvil basado en tecnologías de teleoperación open-source para la toma de acciones de seguridad preventivas e inmediatas en casos de accidentes en minas subterráneas.* Universidad Ricardo Palma, Lima, Perú : 2019.
9. **CÓNDOR TINOCO, Enrique Edgardo.** *Algoritmos resueltos con python.* Bogota : EIDEC, 2020.
10. **DOWNEY, Allen; ELKNER, Jeffrey; MEYERS, Chris.** *Aprenda a pensar como un programador con Python.* Buenos Aires : s. n., 2002. ISBN 0-9716775-0-6.
11. **RAMÍREZ VIQUE, Robert.** *Métodos para el desarrollo de aplicaciones móviles.* Universidad Oberta de Catalunya, Catalunya, España : 2018.

12. **TORRES REMÓN, Manuel Ángel.** *Diseño web con HTML5 y CSS3.* Lima : Macro EIRL, 2018. Primera edición.
13. **Tecnología para desarrolladores Web.** MDN Web Docs. [En línea] 13 de octubre de 2021. [Citado el: 15 de octubre de 2021.] <https://developer.mozilla.org/es/docs/Web/JavaScript>.
14. **JIMENÉS HERNÁNDEZ, Alfonso; AGUILAR PÉREZ, Javier Manuel.** *Diseño y desarrollo de una aplicación Android para el reconocimiento óptico de caracteres.* Universidad de Valladolid, Valladolid, España : 2018.
15. **Grupo Asesor.** *Diagrama de flujo.* Buenos Aires : s.n., 2015.
16. **VARGAS CORDERO, Zoila Rosa.** *La investigación aplicada: una forma de conocer las realidades con evidencia.* [En línea] 15 de diciembre de 2009. [Citado el: 1 de octubre de 2020.] <https://n9.cl/0cekh>
17. **GENMA, G. Juanes.** *Cuadernos de Seguridad.* [En línea] 18 de junio de 2021. [Citado el: 5 de agosto de 2021.] <https://cuadernosdeseguridad.com/2021/06/reconocimiento-de-matriculas-zkteco/>.
18. **ÁLVARES GUTIÉRREZ, Edwin Leonel; JIMÉNES LÓPEZ, Fabián Rolando.** *Generación de Mapa Global 2D y SLAM usando LiDAR y una Estéreo Cámara para el seguimiento de movimiento de un robot móvil.* Iteckne, Tunja, Colombia : 2019.
19. **PRIETO AYLLÓN, Samuel Antonio.** *Sistema robotizado móvil para digitalización de interiores de edificios.* Universidad de Castilla-La Mancha, Ciudad Real, España : 2019.
20. **CHOQUEHUANCA HUARACCALLO, Ronald; LLAIQUI LOBON, Luis Miguel.** *Robot móvil para detección de personas en lugares inaccesibles frente a desastres naturales en la ciudad Arequipa: caso sismos.* Universidad Católica de Santa María, Arequipa : 2020.
21. **CRISÓSTOMO POMA, Jhimmy Jhoel.** *Diseño de un robot móvil de servicio para aplicaciones de fumigación del cultivo de maíz en la provincia de Concepción de la región Junín.* Universidad Continental, Huancayo-Perú : 2020.
22. **TOMÁS GIRONÉS, Jesus.** *El Gran Libro de Android.* Valencia : Marcombo, 2016. Quinta Edición.

23. **Fundation OpenJs.** Node.js® . [En línea] 16 de agosto de 2021. [Citado el: 15 de octubre de 2021.] <https://nodejs.org/en/about/>
24. \_\_\_\_\_. ExpressJs. [En línea] 10 de junio de 2021. [Citado el: 15 de octubre de 2021.] <https://expressjs.com/es/>

## **ANEXOS**

```

import cv2 #libreria cv2
import pyesseract #libreria pyesseract
import webbrowser as wb #libreria webbrowser
import tkinter as tk #libreria tkinter

#Inicio de video en vivo
cuadro=tk.Tk()
ancho_cam,alto_cam=688,400
cap=cv2.VideoCapture(0)
cap.set(1,ancho_cam)
cap.set(4,alto_cam)

#Files
file="db.txt"
file1="guias.txt"

#Colores
fondo_entrar="#000000" #color azul
fondo_salir="#FF0000" #color rojo
fondo_correcto="#008000" #color verde
fondo_incorrecto="#FF0000" #color rojo
fondo_entrada="#000000" #color negro

#Creacion de ventana principal
ventana=tk.Tk() #Creacion de ventana para visualizar los datos
ventana.title("Aplicativo Panel Naranja") #Titulo de la ventana
ventana.geometry("500x500+50+50") #Tamaño de la ventana
ventana.resizable(width=False,height=False)
ventana.configure(background=fondo_entrada) #Color de fondo de la ventana

#Variables almacenadas de los cajetines de texto
variable=tk.StringVar()

Microsoft PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.
Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6
PS C:\Users\User\Desktop>

```

Figura 51. Programa utilizado para el desarrollo del aplicativo Visual Studio Code



Figura 52. Guía GRE200



**Figura 53. Número ONU 1075 (GLP) utilizado en las pruebas**



**Figura 54. Nivel advertencia brindando información obtenida del Aplicativo hacia el nivel Operaciones**



**Figura 55. Nivel Operaciones interviniendo una fuga de un “Mat-Pel” con los datos brindados del nivel Advertencia**



**Figura 56. Piloto especializado del dron**



**Figura 57. Evidencias del día de pruebas con apoyo de la compañía de bomberos César A. Habich Sosa N.º 30**