

ANEXOS DEL TRABAJO: CARACTERIZACIÓN EN EL ESPECTRO VISIBLE E INFRARROJO DE CULTIVOS AGRÍCOLAS DESDE UN VEHÍCULO AÉREO MULTIRROTOR



Julián Andrés Bolaños Ortega

Trabajo de grado en Ingeniería Electrónica y Telecomunicaciones

Director:
Ing. Liseth Viviana Campo

Co Director:
PhD. Juan Carlos Corrales Muñoz
PhD(c). Edgar Leonairo Pencue Fierro

Universidad Del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Popayán, 2017

Julián Andrés Bolaños Ortega

Anexos del trabajo: CARACTERIZACIÓN EN EL
ESPECTRO VISIBLE E INFRARROJO DE CULTIVOS
AGRÍCOLAS DESDE UN VEHÍCULO AÉREO
MULTIRROTOR

Trabajo de grado presentado en la Facultad de Ingeniería
Electrónica y Telecomunicaciones de la
Universidad del Cauca para la obtención del
Título de

Ingeniero en:
Electrónica y Telecomunicaciones

Director:
Ing. Liseth Viviana Campo

Co Director:
PhD. Juan Carlos Corrales Muñoz
PhD(c). Edgar Leonairo Pencue Fierro

Popayán
2017

Contenido

Contenido.....	I
Lista de figuras.....	V
Lista de Tablas	VII
Anexo A.....	1
Estado del arte.....	1
A1. Índices de vegetación con fotogrametría.....	1
A2. Aplicaciones de los UAV en la agricultura de precisión.....	3
A3. Información adicional sobre realización del vuelo.....	4
A4. Plataformas software para el análisis de fotografías aéreas	5
Anexo B.....	7
Selección de herramientas	7
B1. Dispositivos y herramientas	7
B1.1. Características del ambiente de desarrollo.....	7
B1.2. Librerías	8
B1.3. Lenguajes de programación.....	9
B1.4. Entornos de desarrollo	9
B1.5. Servicios de Nube Microsoft Azure.....	10
B1.6. Herramientas de ayuda	10
B2. Selección de herramientas de hardware.....	12
B2.1. Aeronaves No tripuladas	12
B2.2. Herramientas de Procesamiento.....	14

B2.2.1. Tarjeta de desarrollo Raspberry Pi B+ y cámara	14
Anexo C	15
Interfaz gráfica prototipo	15
C1. Prototipo Móvil	15
C1.1. Vista Principal	15
C1.2. Vista Proceso	16
C1.3. Vista Aeronaves	16
C1.4. Vista Cámaras	18
C1.5. Vistas Crear Proceso	18
C1.6. Vista Resultados	20
C1.7. Vista Iniciar Sesión	20
C2. Prototipo Web	21
Anexo D	27
Implementación Prototipo	27
D1. Aplicación Web	27
D1.1. Managed Beans o Beans Administrados	27
D1.2. Conexión a Bases de Datos	28
D2. Aplicación Móvil	31
D2.1. Modelos	31
D2.2. Activitys o actividades	35
D2.3. Adapters	40
Anexo E	43
Diagramas Prototipo	43
E1. Vista lógica:	43
E1.1. Diagrama de clases	43
E1.2. Diagramas de secuencia	44
E2. Vista de desarrollo:	45
E2.1. Diagrama de Componentes	45
E2.2. Diagrama de Paquetes	46
E3. Vista de Procesos:	47

E4. Vista Física:	47
E4.1. Obtención de la arquitectura del sistema	47
E4.2. Diagrama de despliegue	48
E5. Vista de escenarios:	49
E5.1. Casos de uso del prototipo	49
E5.2. Casos de Uso Detallados del Prototipo	50
Anexo F	55
Anexos pruebas finales	55
F3. Datos completos pruebas finales	58
F3.1. Filtro Rojo #19 con AWB 11:	58
F3.2. Filtro Rojo #19 con AWB 0.54,1:	60
F3.3. Filtro Azul #2007 con AWB 11:	61
Anexo G	65
Manual de configuración Prototipo	65
G1. Instalación aplicación móvil	65
G1.1. Descarga APK	65
G2. Acceso al servicio Web	66
Bibliografía	67

Lista de figuras

Figura 1. Vista software PUTTY	11
Figura 2 Interfaz de VNC.....	12
Figura 3. AR Drone Parrot 2.0.....	13
Figura 4. Aeronave Phantom 3 Standard	14
Figura 5. Raspberry Pi 2.0 y cámara.....	14
Figura 6 Vista principal.....	15
Figura 7 Vista Proceso.....	16
Figura 8 Vista mis aeronaves y agregar nuevas aeronaves.....	17
Figura 10 Vista cámaras y agregar cámaras.....	18
Figura 12 Pasos Nuevo Proceso.....	19
Figura 13 Vista Resumen Nuevo Proceso	19
Figura 14 Vista Resultados Proceso	20
Figura 15 Vista iniciar sesión.....	20
Figura 16 Vista registro.....	21
Figura 17 Pantalla de bienvenida.....	21
Figura 18 Pantalla de Inicio de sesión.....	22
Figura 19 Pantalla de inicio principal.....	22
Figura 20 Pantalla de Proceso	22
Figura 21 Información de un proceso.....	23
Figura 22 Pantalla subir fotografías.....	23
Figura 23 Pantalla subiendo fotografías.....	23
Figura 24 Pantalla subir fotografías deshabilitada.....	24
Figura 25 Pantalla Subir Fotografías.....	24
Figura 26 Pantalla proceso mostrar resultados	24
Figura 27 Ejemplo de ventana resultados AppWeb	26
Figura 28 Clases Beans Administrados	28
Figura 29 EJB Proceso.....	29
Figura 30 EJB Usuario	29
Figura 31 EJB Sub Proceso	29
Figura 32 Clase Handler Proceso	30
Figura 33 Handler Sub Proceso	30
Figura 34 Handler Usuario	31
Figura 35 Implementación Handler Proceso	31

Figura 36 Clase Usuario	32
Figura 37 Clase Proceso	33
Figura 38 Clase Sub Proceso	33
Figura 39 Clase Cámara	34
Figura 40 Clase Aeronave	34
Figura 41 Clase Main Activity	35
Figura 42 Clase Actividad Nuevo Proceso	36
Figura 43 Clase Actividad Proceso.....	37
Figura 44 Actividad Cámaras	37
Figura 45 Clase Actividad Añadir Cámara.....	38
Figura 46 Actividad Aeronaves.....	38
Figura 47 Clase Actividad Añadir Aeronave	39
Figura 48 Clase Actividad Login	39
Figura 49 Clase Actividad Registro	40
Figura 50 Clase Adapter Aeronave	40
Figura 51 Clase Adapter Procesos.....	41
Figura 52 Clase Adapter Sub Proceso	41
Figura 53 Clase Adapter Cámara	41
Figura 54 Diagrama de Clases	44
Figura 55 Diagrama de secuencia.....	45
Figura 56 Diagrama de Componentes.....	46
Figura 57 Diagrama de Paquetes.....	47
Figura 58 Diagrama de actividad.....	47
Figura 59 Arquitectura del sistema	48
Figura 60 Diagrama de despliegue.....	49
Figura 61 Diagrama de Casos de uso	49
Figura 62 Resultados de la captura Filtro Rojo AWB 11	55
Figura 63 Resultados NDVI Captura Yaguará.....	56
Figura 64 Sobrevuelo Villamoreno Nariño Cultivo de Arbeja.....	56
Figura 65 Sobrevuelo San Fernando Nariño Cultivo de Arbjea	57
Figura 66 Sobrevuelo Yaguará Huila Cultivo de Arroz	57
Figura 67 Sobrevuelo Campus Universitario	58
Figura 68 Vista iniciar sesión.....	66

Lista de Tablas

Tabla 1. Índices de vegetación más comunes [4], [1].....	3
Tabla 2 Resultados Tarea 1 Filtro rojo awb 11	59
Tabla 3 Resultados Tarea 2 Filtro rojo awb 11	59
Tabla 4 Resultados Tarea 3 Filtro rojo awb 11	60
Tabla 5 Resultados Tarea 1 Filtro azul AWB 0.54,1.....	60
Tabla 6 Resultados Tarea 2 Filtro azul AWB 0.54,1.....	61
Tabla 7 Resultados Tarea 1 Filtro azul AWB 0.54,1.....	61
Tabla 8 Resultados Tarea 1 Filtro azul AWB 11.....	62
Tabla 9 Resultados Tarea 2 Filtro azul AWB 11.....	62
Tabla 10 Resultados Tarea 2 Filtro azul AWB 11.....	63

Anexo A

Estado del arte

Para este trabajo de grado se realiza una revisión que contempla un análisis de los índices de vegetación, posteriormente, se estudian aplicaciones de las aeronaves no tripuladas en la agricultura de precisión y finalmente se concluye con algunas aplicaciones o plataformas ya desarrolladas que están dirigidas a la agricultura de precisión y que contemplan tecnología de aeronaves no tripuladas.

A1. Índices de vegetación con fotogrametría

Uno de los aspectos principales entorno a la AP es el estudio de los índices de vegetación puesto que con ellos se tiene un conocimiento del estado vegetativo de las plantas. Una correcta interpretación se puede traducir en beneficios económicos gracias al ahorro de fertilizantes y reducción de las zonas de influencia de los tratamientos realizados. Para poder desarrollar la propuesta se realiza una exploración en la que se encontró que el trabajo [1] estudia la biomasa en un cultivo a partir de los índices de vegetación NDVI, SAVI, MSAVI, OSAVI, GnyLi, GRVI, MGRVI, RGBVI que se calculan mediante imágenes en el espectro visible e infrarrojo cercano obtenidas desde un UAV octacóptero, el objetivo es encontrar relaciones entre los factores físicos de los cultivos y las imágenes capturadas. El trabajo [2] se basa en la comparación de resultados teniendo en cuenta el tamaño de los píxeles, se busca evaluar la exactitud de los índices de vegetación NDVI, GNDIV, NGRDI, TGI obtenidos a partir de imágenes en el espectro visible e infrarrojo cercano. El método utilizado se basa en software libre con el inconveniente de que al incrementar la cantidad de imágenes y el tamaño del cultivo crece exponencialmente el esfuerzo y trabajo requerido para obtener los resultados. En el trabajo de [3] se encuentran 6 índices de vegetación ExG, NGRDI, CIVE, VEG, WI, ExGR a partir de imágenes en RGB. Para el análisis se hacen cambios en la altura y en el tiempo de la captura de datos desde un cuadricoptero. Entre la información de cada imagen se tiene datos de GPS y los grados de libertad “*row, pitch, yaw*” del cuadricoptero. En [4] se plantea determinar el tipo de vegetación en un cultivo basándose en un método para la determinación de vegetación denominado Otsu que verifica en cual pixel existe vegetación y en cual no [5]. La aeronave es un cuadricoptero que vuela a una altura aproximada de 30m capturando imágenes en

el espectro RGB e infrarrojo, se utilizan los índices de vegetación ExG y NDVI con el fin de diferenciar entre el suelo con vegetación y suelo libre. En el trabajo [6] se hace uso de los índices de vegetación SAVI y RVI para evaluar la concentración de nitrógeno en las hojas de las plantas de trigo y de arroz, los esfuerzos realizados se basan en analizar el comportamiento de las diferentes bandas espectrales y se concluye cuales índices de vegetación son apropiados para determinar la absorción de nitrógeno en las hojas. En [7] se desea encontrar la altura del follaje así como el área de biomasa dentro de un cultivo de maíz utilizando imágenes obtenidas a partir de una aeronave no tripulada de bajo costo. El lugar de realización del trabajo es 84km al norte de Beijing, China a 30 metros sobre el nivel del mar. Se realiza un vuelo a una altura de 150 metros a una temperatura de 9.1°C capturando imágenes RGB con un 80% de traslape longitudinal y un 40% de traslape lateral, se utilizan índices de vegetación encontrados en el rango del espectro visible. La información de la altura del follaje y la biomasa se presentan en un mapa georreferenciado obtenido a partir de los índices de vegetación. En [8] se describen algoritmos para la detección de la fracción vegetal en un cultivo a partir de imágenes multiespectrales capturadas a diferentes tiempos para medir la reflectancia espectral durante el crecimiento de la planta, se hace uso de los índices de vegetación NDVI, GNDVI, VARI para posteriormente relacionarlos con la fracción vegetal existente, los resultados se presentan en forma de gráficos de porcentaje de fracción vegetal con respecto a cada uno de los índices de vegetación estudiados. En [9] se desea establecer técnicas y métodos para una correcta segmentación de imágenes basándose en los cultivos de maíz y cebada, la investigación tienen en cuenta diferentes índices de vegetación además del algoritmo Otsu [5], todo con el fin de diferenciar entre las plantas, el suelo y el cielo. Los resultados presentados en los que se utiliza como herramienta de procesamiento Matlab demuestran que es posible determinar y diferenciar exactamente los suelos con vegetación y además diferentes tipos de la misma. No se utiliza un vehículo aéreo multirrotor.

En la tabla 1 se presenta una breve descripción de los índices de vegetación más importantes encontrados.

Índice de vegetación	Descripción	Formula
NDVI	El Índice de vegetación de diferencia normalizada sirve para medir el crecimiento y calidad de la vegetación. Se calcula a partir de las bandas de infrarrojo cercano y el espectro visible.	$\frac{R_{850} - R_{680}}{R_{850} + R_{680}}$
SAVI	El índice de vegetación ajustado al suelo está diseñado para disminuir la alteración en el NDVI que se produce en zonas áridas.	$(1 + L) \frac{R_{800} - R_{670}}{R_{800} + R_{670} + L}$
MSAVI	Índice de vegetación modificado ajustado al suelo	$\frac{2R_{800} + 1 - \sqrt{(2R_{800} + 1)^2 - 8(R_{800} - R_{670})}}{2}$
OSAVI	Índice de vegetación ajustado al suelo óptimo	$(1 + 0.16) \frac{R_{800} - R_{670}}{R_{800} + R_{670} + 0.16}$
GnyLi	Índice de Gny and Li	$\frac{R_{900}R_{1050} - R_{955}R_{1220}}{R_{900}R_{1050} + R_{955}R_{1220}}$
GRVI	Índice de vegetación verde y rojo	$\frac{R_{590} - R_{680}}{R_{590} + R_{680}}$
MGRVI	Índice de vegetación de verde y rojo modificado	$\frac{(R_{590})^2 - R_{680}^2}{(R_{590})^2 + R_{680}^2}$
RGBVI	Índice de vegetación Azul, verde y rojo	$\frac{(R_{590})^2 - R_{680} * R_{515}}{(R_{590})^2 + R_{680} * R_{515}}$
GNDVI	Índice de vegetación de diferencia normalizada verde	$\frac{R_{850} - R_{550}}{R_{850} + R_{550}}$
ExG	Índice con exceso de verde	$\frac{2R_{590} - R_{680} - R_{515}}{R_{590} + R_{680} + R_{515}}$
CIVE	Índice de extracción del color	$\frac{-0.881R_{590} + 0.441R_{680} + 0.441R_{515}}{R_{590} + R_{680} + R_{515} + 18.78745}$
WI	Índice de Woebbecke	$\frac{R_{590} - R_{515}}{R_{680} - R_{590}}$
ExGR	Exceso de verde menos exceso de rojo ExG-ExR	$ExG - \frac{-R_{590} + 1.4R_{680}}{R_{590} + R_{680} + R_{515}}$

Tabla 1. Índices de vegetación más comunes [4], [1].

A2. Aplicaciones de los UAV en la agricultura de precisión

Se pueden tener diversas aplicaciones con las imágenes obtenidas a partir de las aeronaves no tripuladas, en [10] se encuentra el tamaño de la cubierta vegetal a partir del análisis de fotografías aéreas, el estudio se centra en cultivos de trigo para poder obtener un análisis de la deformación en dichas plantaciones denominada "Lodging", los datos se obtienen desde un globo a una altura entre 50 y 150 metros en dos parcelas ubicadas en diferentes lugares, comparando los resultados a lo largo del tiempo intentando mantener las condiciones para cada prueba. Las imágenes obtenidas se pueden emplear para diferentes propósitos tales como las

de crear mapeos 2D y 3D, construcción de modelos digitales de terreno, realizar medidas en el follaje, conteo de plantas, obtención de mosaicos geo-referenciados entre otros, que brindan a los agricultores información completa de sus cultivos con una buena relación costo beneficio [11] [12]. En [13] se realiza la medición de altura de las plantas en un cultivo de maíz utilizando tecnología LiDAR desde un hexacóptero. Las medidas se comparan con las reales para verificar los resultados obtenidos. En [14] se estudia un método denominado OBIA para realizar el mapeo de malezas en un cultivo de maíz en España utilizando UAVs, para llevar a cabo dicho propósito se utiliza una cámara multispectral de 6 bandas en las que se incluyen rangos del espectro de infrarrojo cercano y visible. La aeronave utilizada es un cuadricóptero que puede volar manual u autónomamente, los resultados obtenidos al utilizar este algoritmo fueron satisfactorios, el uso de esta tecnología ayudó a disminuir el uso de herbicidas para controlar las malezas presentes en el cultivo y cumplir con los requisitos españoles de aplicación de herbicidas. En [15] se pretende desarrollar un método para detectar la maleza "Sircium Arvense" de las plantaciones de remolacha azucarera, se utiliza una cámara RGB comercial y se hace uso de dos índices de vegetación en el espectro visible ExG y RB (índice rojo y azul), al realizar algunas operaciones entre ellos se encuentra que se puede identificar dicha maleza en los cultivos.

A3. Información adicional sobre realización del vuelo

Las siguientes relaciones son utilizadas para obtener información del vuelo a realizar.

- Número de fotografías por cada línea de vuelo:

$$NFLV = \frac{\text{anchoTerreno}}{B} \quad (1)$$

- Número de líneas de vuelo es:

$$NLV = \frac{\text{largoTerreno}}{A} \quad (2)$$

- El número total de fotografías (NTF) aproximado es:

$$NTF = NLV * NFLV \quad (3)$$

- El tiempo total de la misión es:

$$TM = NTF * T_o \quad (4)$$

A continuación, a partir de (1), (2), (3), (5), (7), (12), (13), (14) y (15) se obtiene la relación entre la altura y el área total del terreno, teniendo en cuenta el tiempo de vuelo determinado para la misión (TM).

$$TM = NTF * To = (NLV * NFLV) * To$$

$$TM = \left(\frac{\text{anchoTerreno}}{B} * \frac{\text{largoTerreno}}{A} \right) * to$$

$$TM = \frac{\text{ÁreaTerreno} * To}{\text{ÁreaSensor} \left(\frac{h}{f} \right)^2 \left(1 - \frac{v}{100} \right) \left(1 - \frac{u}{100} \right)} \quad (5)$$

Reemplazando (15) en (9)

$$V_{aeronave} = \frac{L_2 \left(1 - \frac{v}{100} \right)}{t_0}$$

$$V_{aeronave} = \frac{(dHs) * \sqrt{\frac{\text{áreaTerreno} * To}{\text{área sensor} * TM \left(1 - \frac{v}{100} \right) \left(1 - \frac{u}{100} \right)}}}{t_0} \left(1 - \frac{v}{100} \right)$$

$$V_{aeronave} = (dHs) * \sqrt{\frac{\text{áreaTerreno} * \left(1 - \frac{v}{100} \right)}{\text{área sensor} * TM * \left(1 - \frac{u}{100} \right) * To}} \quad (6)$$

Se puede apreciar que según la relación (6) se tiene una **proporcionalidad directa** entre el área del **terreno a cubrir** y la **velocidad de la aeronave**.

A4. Plataformas software para el análisis de fotografías aéreas

En el estudio de trabajos realizados es importante resaltar las herramientas exploradas que existen en el mercado y que están destinadas a brindar un apoyo a la AP, entre las que se destacan:

- **Aplicación de Sign Post:** Sign Post es una empresa dedicada a desarrollar soluciones basadas en aeronaves no tripuladas, los resultados que proporcionan son modelos digitales de superficie, mosaicos georreferenciados con los que se monitorea el crecimiento y se realiza una clasificación de las plantas, y haciendo uso de imágenes multispectrales se determina estrés de las plantas, clorofila, sequía, indicador de biomasa, así como también

basándose en dicha información se entregan tablas de concentración de nitrógeno, fósforo y potasio. No se presentan características de despliegue de resultados a partir de aplicaciones especializadas [16].

- **Aplicación de Delair Tech:** Delair-Tech es una empresa dedicada a desarrollar aeronaves no tripuladas, entre sus servicios se encuentra el soporte completo a la AP. Sus aeronaves no tripuladas están equipadas con sensores multiespectrales que sirven para calcular los índices de vegetación más importantes, entre los resultados derivados se encuentran mosaicos georeferenciados, información del estrés de los cultivos, desnutrición y enfermedades, así como también un conteo de plantas. Se especifica que utiliza una interfaz web llamada “*Delair-analytics*” para la visualización de los resultados, y también se especifica que dentro del sistema se utiliza la herramienta Pix4D para el procesamiento de las imágenes recolectadas [17].
- **Sioma:** Sioma es una empresa colombiana dedicada a apoyar la AP en el Urabá antioqueño, la información se captura desde diferentes tipos de sensores ubicados a lo largo de los cultivos y en aeronaves no tripuladas, los resultados se presentan en una aplicación web en la cual se pueden evidenciar índices NDVI, datos del clima, el viento, estadísticas de control de insumos, entre otras [18].

Anexo B

Selección de herramientas

B1. Dispositivos y herramientas

Se describen las herramientas *software* que se usan para la implementación del prototipo móvil y web, así como las tecnologías utilizadas para la implementación del módulo de procesamiento de información.

B1.1. Características del ambiente de desarrollo

En el ámbito de la computación, el ambiente de desarrollo es definido como el conjunto de herramientas de programación, procesos, y componentes *hardware* que se usan en el desarrollo de un producto *software*, en algunos casos también puede incluir las características del espacio físico en donde se efectúa dicho proceso.

En ese orden de ideas, como una herramienta *software* importante para llevar a cabo un control de versiones se usa *Git*. Pues permite la comparación de los códigos entre versiones así como un *backup* de información.

El desarrollo del módulo de despliegue de información es compuesto por dos tipos de tecnologías, una móvil y otra web, para lo cual se deben definir las tecnologías a utilizar.

El desarrollo de la parte móvil se realiza completamente destinada a un sistema operativo Android bajo la herramienta Android Studio, que es proveída por Google y se caracteriza por: llevar a cabo una renderización de interfaces en tiempo real; la construcción de los proyectos está basada en *Gradle* que permite un manejo ágil de dependencias y versiones; también permite realizar un control de versiones de SDK mediante el *SDKManager*, entre otras características. Los motivos para seleccionar este ambiente de desarrollo son:

1. Por ahora es el único ambiente de integrado de desarrollo(IDE) que es oficial dedicado a Android, reemplazando al antes utilizado Eclipse IDE que manejaba el ADT(Android Developer Tools)
2. Permite una buena visualización al diseñar las interfaces gráficas.
3. Permite un buen control de errores al momento de elaborar el código fuente.

Para el desarrollo de la parte web se define como la tecnología empleada Java Server Faces (JSF) sobre el entorno de desarrollo NetBeans. Pues permite una buena agilidad a la hora de realizar diferentes tareas como el subir información, además el lenguaje utilizado en este caso es Java lo que abre la posibilidad de poder reutilizar parte del código utilizado en la aplicación móvil. Se pensaba utilizar en un principio el lenguaje PHP pero debido a problemas con la comunicación hacia los servidores de bases de datos mientras estaba corriendo en sistemas Linux se decide desistir de la idea.

El módulo para el procesamiento de información es realizado en Python que además estará orientado a ser controlado por el sistema Web, que será el encargado de llamar los diferentes scripts de Python.

Las bases de datos se implementan sobre un servicio Móvil brindado por la plataforma de *Microsoft Azure*, esto permite el acceder a la base de datos mediante un *BackEnd* desarrollado en JavaScript. Es importante mencionar que el servidor de base de datos es SQLServer, debido a que es necesario tenerlo en cuenta para la instalación de los Drivers de comunicación.

Por último se establece que todos los componentes serán instalados sobre un servicio de Máquina virtual en la nube corriendo un sistema operativo Ubuntu Server 14.04 LTS ofrecido por Microsoft Azure, en la cual estará instalado el servidor de Java Empresarial Glassfish y el servidor Apache Tomcat, los cuales permitirán el acceso Web público para cada una de las aplicaciones.

B1.2. Librerías

Las librerías utilizadas en el desarrollo del prototipo se describen a continuación.

1. Librería Opencv

Esta librería es utilizada en el sistema para realizar el procesamiento de la información, es multiplataforma y es posible utilizarla en varios lenguajes, es un conjunto de funciones que permiten llevar a cabo procesamiento y tratamiento de imágenes.

2. Librería Primefaces

Esta librería es utilizada por el *framework* JSF para realizar una renderización de interfaces, facilita en gran medida las labores de diseño para sistemas web por lo que será manejado en el prototipo.

3. Librería Matplotlib

Esta librería será la encargada de realizar un mapeo de los datos para lograr un mapa de color, similar a la librería Opencv, puede trabajar con imágenes, pero su uso se centra en el manejo y representación gráfica de cualquier tipo de información.

4. Librería Picasso

Picasso se utiliza en el desarrollo de la aplicación móvil Android, y permite la renderización de imágenes de forma asíncrona, esto evita los problemas de memoria ocasionados con objeto del manejo de imágenes.

5. JDBC Driver para SQLServer

Esta librería se encarga de facilitar la comunicación entre la aplicación web y los servidores de bases de datos de nube.

B1.3. Lenguajes de programación

Java: Es un lenguaje de programación fiable, seguro y rápido. Hay numerosas aplicaciones y sitios web que se basan en java, desde *laptops* hasta centros de datos, desde consolas para video juegos hasta súper-computadoras, desde teléfonos móviles hasta tablets, neveras, equipos doméstico, *Smart watch*, Televisores, java está en todas partes. Probado, perfeccionado y ampliado por toda una comunidad de desarrolladores; diseñado para permitir el desarrollo de aplicaciones portátiles en diferentesplataformas informáticas [19].

Python: Lenguaje de programación orientado a objetos claro y potente. Entre sus características más notables están: Una sintaxis intuitiva, ésta permite una fácil interpretación del código; es fácil de usar, esto permite que la programación sea rápida; es multiplataforma y de fácil extensión. Lenguaje de programación multiparadigmático, esto permite que los desarrolladores programen en diferentes estilos: Orientado a objetos, imperativa y programación estructurada [20].

B1.4. Entornos de desarrollo

Android Studio: Es el entorno escogido para el desarrollo de la aplicación móvil, actualmente es la plataforma principal para el desarrollo de aplicaciones bajo el sistema operativo Android de forma nativa.

NetBeans: Es un entorno de desarrollo de código abierto. No tiene un lenguaje de programación en específico, soporta diferentes tecnologías entre ellas Java, PHP, Groovy, HTML5. Puede instalarse en varios sistemas operativos entre los que se encuentran Windows y Linux [21].

Sus características principales son: 1) Asistencia para la creación y configuración de proyectos. 2) Editor de código, multilenguaje; proporciona sugerencias de código, control de versiones, localización de clases, comprobaciones sintácticas. 3) Gestión de grandes proyectos con el uso de diferentes vistas. 4) Depurador de errores para encontrar fallos en el código definiendo puntos de ruptura. 5) Servidores de aplicaciones, permite gestionar desde el propio IDE diferentes servidores de aplicación que serán de utilidad a la hora de realizar pruebas (Apache Tomcat, GlassFish, JBoss) [21].

B1.5. Servicios de Nube Microsoft Azure

Servicio Móvil: Es un servicio que ofrece Microsoft Azure el cual permite realizar implementaciones de servicios de bases de datos en la nube, la característica más importante es que proporciona un Back End en Java Script el cual es consumido por cualquier aplicación móvil.

Máquinas virtuales: Es un servicio ofrecido por Microsoft Azure el cual permite realizar implementaciones de instancias de sistemas operativos, es posible implementar imágenes de Linux o Windows de forma intuitiva y rápida. La característica más importante es que proporciona una IP pública permitiendo acceder a las aplicaciones desarrolladas con la única condición de tener internet.

B1.6. Herramientas de ayuda

Putty-Connection: Es una herramienta que permite realizar conexión remota mediante SSH a equipos que corran sistema operativo Linux, se utiliza para la comunicación con las máquinas virtuales de Windows Azure y la tarjeta Raspberry Pi, para realizar la conexión, el único requerimiento es que el dispositivo a conectarse deberá estar disponible en la red. La interfaz principal se presenta en la **¡Error! No se encuentra el origen de la referencia..**

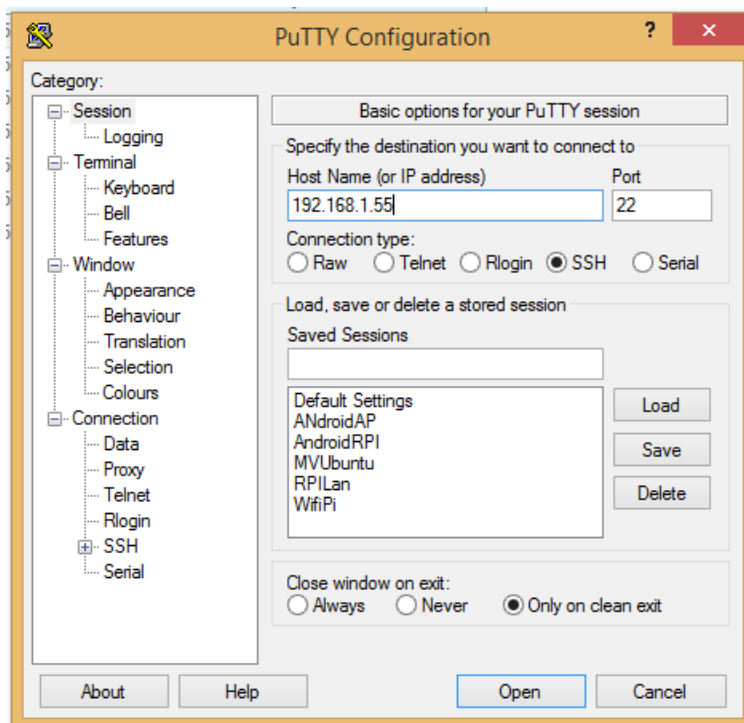


Figura 1. Vista software PUTTY

TightVnc: Este software es el responsable de permitir acceso remoto al escritorio bien sea de la tarjeta Raspberry Pi o de la máquina virtual de Microsoft Azure, en la raspberry es útil para facilitar el copiado de fotografías y el control de carpetas, en la máquina virtual fue utilizado para instalar los entornos donde se despliega la aplicación Web tales como el servidor GlassFish. En la *Figura 2* Se puede encontrar la interfaz de conectividad de VNC, en ella se proporciona la IP ya sea pública o privada del host a conectar, la conexión se realizará con éxito siempre y cuando el servicio de VNC esté instalado e iniciado en el host.

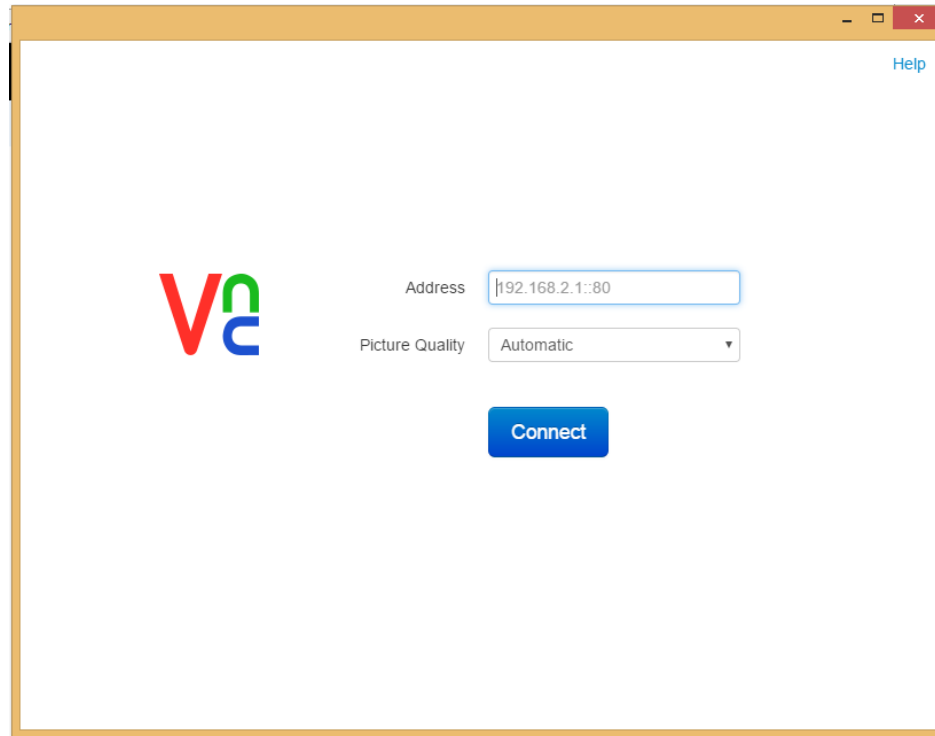


Figura 2 Interfaz de VNC

JuiceSSH: Este software es una aplicación móvil popular la cual permite realizar la conexión a la tarjeta Raspberry Pi, se utilizó en las pruebas finales para permitir mayor agilidad en las mismas.

Microsoft XRDP: Se utiliza como alternativa a VNC para acceder al servidor de escritorio remoto, tiene la desventaja que la aplicación móvil ya no funciona para acceder a sistemas Ubuntu.

B2. Selección de herramientas de hardware

Los criterios principales en la selección de Hardware es el costo de estos, y su funcionalidad, en este trabajo se realizaron pruebas con dos tipos de plataformas aéreas no tripuladas descritas a continuación.

B2.1. Aeronaves No tripuladas

- **Parrot Ar Drone 2.0**

Este dron se caracteriza por contar con 2 cámaras, una vertical con la que se adquiere información de posicionamiento y otra horizontal que permite la adquisición de fotografías HD de 720p 30 FPS. Este vehículo aéreo no

tripulado cuenta además con las siguientes especificaciones técnicas, y herramientas. [22]

- Peso con carena interior: 380 g
- Peso Con carena exterior: 420 g
- Wi-Fi: Wi-Fi b g n
- Giroscopio: 3 ejes, precisión de 2000°/segundo
- Acelerómetro: 3 ejes, precisión de ± 50 mg
- Magnetómetro: 3 ejes, precisión de 6°
- Sensor de presión: Precisión de ± 10 Pa
- Sensores de ultrasonidos para medir la altitud: Medición de la altitud
- Cámara vertical: QVGA 60 FPS para medir la velocidad en vuelo
- Procesador: Procesador 1 GHz 32 bits ARM Cortex A8 con DSP vídeo 800 MHz TMS320DMC64x
- RAM: DDR2 1 GB a 200 MHz
- USB: USB 2.0 de alta velocidad para las extensiones
- 4 motores sin escobillas de tipo "inrunner": 14,5 vatios y 28 500 rpm
- Engranajes Nylatron: Sí



Figura 3. AR Drone Parrot 2.0

- **Phantom 3 Standard**

Este drone fue el que presento más funcionalidad en este trabajo gracias a las ventajas técnicas y de operatividad que cuenta en comparación al Parrot Ar Drone 2.0, en seguida se describen sus características. [23].

- **Peso con batería y propulsores:** 1216g
- **Max. Velocidad de ascenso:** 5 m/s
- **Max. Velocidad de descenso:** 3 m/s
- **Max. Velocidad:** 16m/s (sin viento).
- **Max. Tiempo de vuelo:** Aprox 25 min.
- **Cámara:** 12 Megapíxeles.

- **Grabación de video:** UHD [2.7k/30p(2704xx1520)]; FHD [1920x1080p (24/25/30)]; HD [1280x720p (24/25/30/48/50/60)].
- **Batería:** 2600mAh Lipo 18650.
- **Puerto de salida video:** Wi-Fi.
- **Puerto de carga:** USB.
- **Frecuencia de operación:** 2.400Ghz-2.483Ghz.
- **Soporte para:** teléfono inteligente.



Figura 4. Aeronave Phantom 3 Standard

B2.2.Herramientas de Procesamiento

B2.2.1. Tarjeta de desarrollo Raspberry Pi B+ y cámara

- **Procesador:** Broadcom BCM2835 SoC full HD (igual que el modelo B).
- **RAM:** 512 MB SDRAM 400 MHz (igual que el modelo B).
- **Almacenamiento:** tarjeta microSD (en el modelo B era una SD).
- **USB:** cuatro puertos USB 2.0 (en el modelo B sólo había 2).
- **Energía:** 600mA hasta 1.8A a 5V (en el modelo B, 750mA hasta 1.2A a 5V).
- **Pines GPIO:** 40 (en el modelo B, 26).
- **Peso:** 45gr.
- **Salida de video:** HDMI 1.4 @ 1920x1200 pixeles.

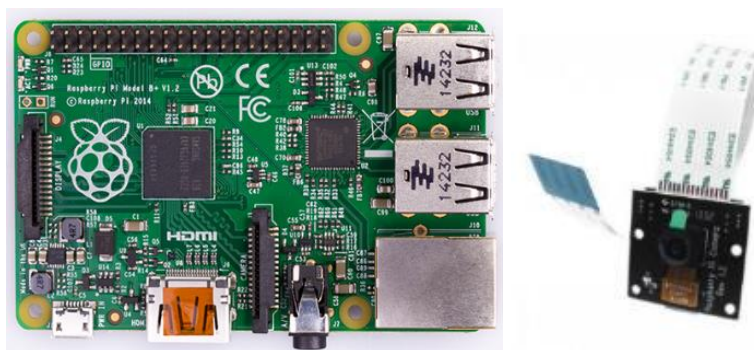


Figura 5. Raspberry Pi 2.0 y cámara

Anexo C

Interfaz gráfica prototipo

C1. Prototipo Móvil

Se presentan las interfaces gráficas completas del prototipo móvil

C1.1. Vista Principal



Figura 6 Vista principal.

C1.2. Vista Proceso



Figura 7 Vista Proceso.

C1.3. Vista Aeronaves

En esta vista son mostradas todas las aeronaves que el usuario tiene agregadas, así como también se muestra un botón flotante como opción para agregar nuevas aeronaves. Las aeronaves mostradas son independientes para cada usuario. Es una de las plantillas de datos que existe.

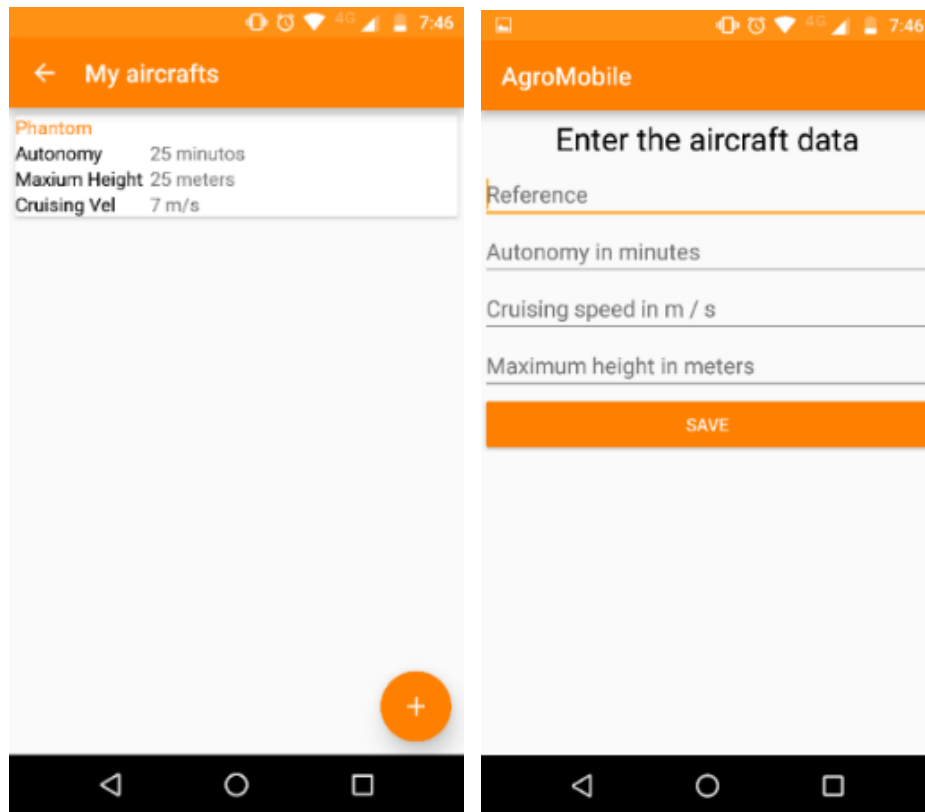


Figura 8 Vista mis aeronaves y agregar nuevas aeronaves

La vista agregar aeronaves recibe los parámetros que caracterizan a cada plataforma y los agrega a la base de datos.

C1.4. Vista Cámaras

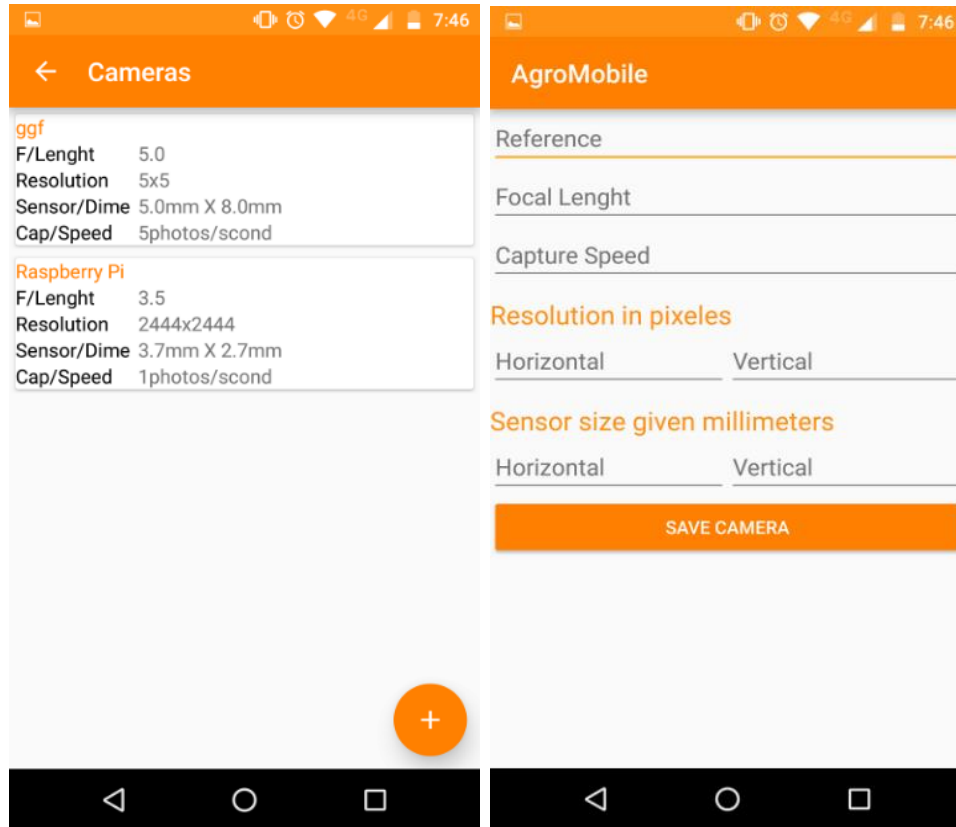


Figura 9 Vista cámaras y agregar cámaras

Esta vista muestra cada una de las cámaras agregadas al sistema, así como también la posibilidad de agregar nuevas cámaras. Las cámaras en el sistema no corresponden explícitamente a un usuario y esto se hace con el fin de ahorrar el trabajo de agregar siempre las mismas cámaras.

Esta vista se encarga de agregar las cámaras a la base de datos general, en ella se ingresan las características principales de cada una.

C1.5. Vistas Crear Proceso

Estas vistas muestran los pasos en los cuales se carga información al sistema, se selecciona en el paso 2 la cámara y el paso 3 la aeronave.

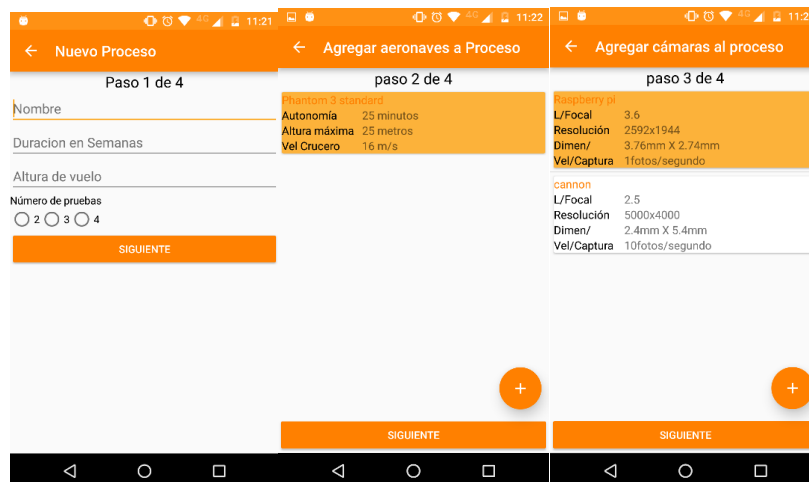


Figura 10 Pasos Nuevo Proceso

Finalmente se puede notar un resumen de información de vuelo donde se destaca la información de vuelo que será utilizada como parámetros para llevar a cabo la misión.

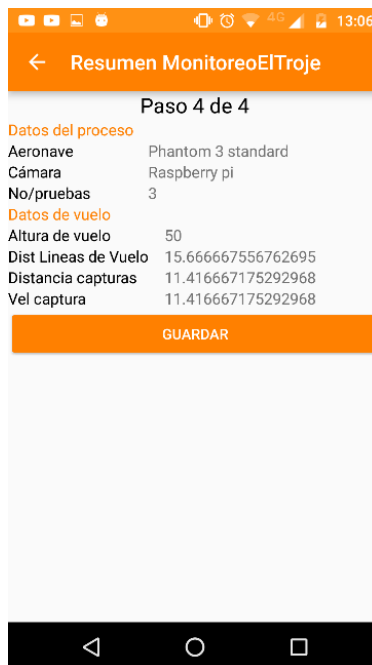


Figura 11 Vista Resumen Nuevo Proceso

C1.6. Vista Resultados

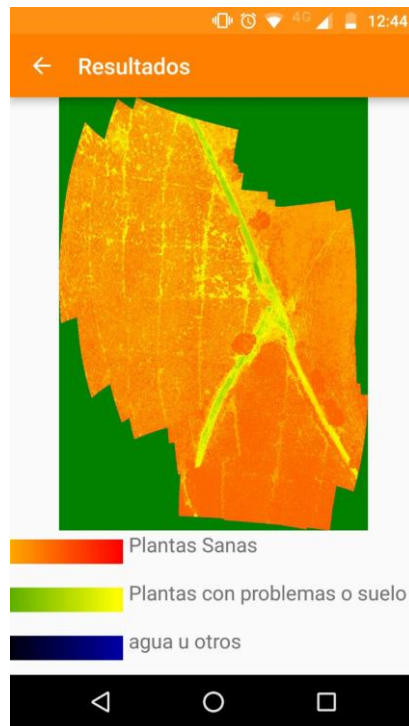


Figura 12 Vista Resultados Proceso

C1.7. Vista Iniciar Sesión

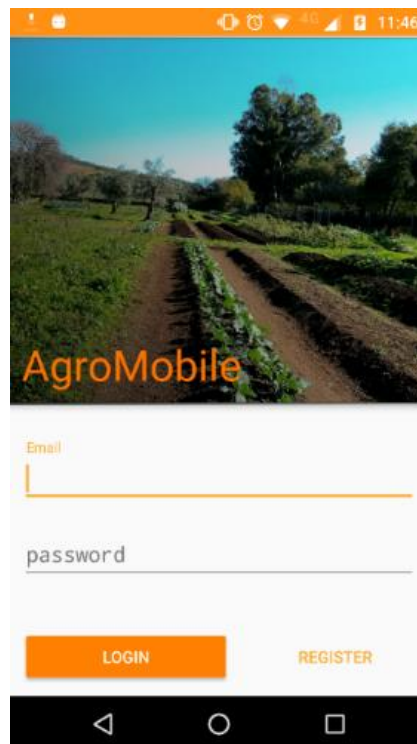


Figura 13 Vista iniciar sesión

En esta vista se presenta el acceso a la aplicación, así como también un botón para habilitar la ventana de registro.

C1.8. Vista Registro

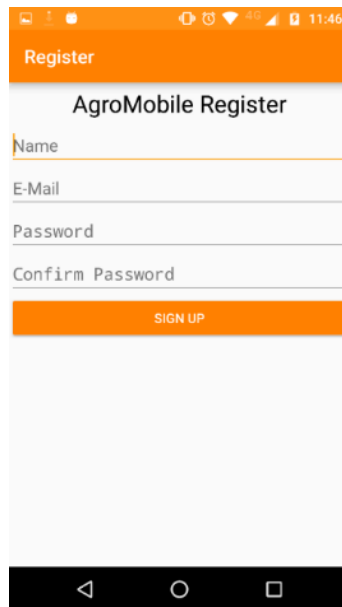


Figura 14 Vista registro.

La vista registro se encarga de permitir crear un nuevo usuario para acceder al sistema.

C2. Prototipo Web

Se presentan las interfaces gráficas completas del prototipo web

C2.1. Interfaz de inicio de sesión de bienvenida



Figura 15 Pantalla de bienvenida.



Figura 16 Pantalla de Inicio de sesión

C2.2. Interfaz de inicio principal

Proceso filtroblue			
Nombre:	filtroblue		
Sub Proceso Actual	3		
Lista de Sub Procesos filtroblue			
Nombre	Fecha de realización	Estado	
Tarea 1 de filtroblue	22/05/2017 11:25:56	Finalizado	ρ
Tarea 2 de filtroblue	22/05/2017 11:25:56	Finalizado	ρ
Tarea 3 de filtroblue	22/05/2017 11:25:56	Finalizado	ρ
Volver		Mostrar Resultados	

Figura 17 Pantalla de inicio principal

C2.3. Interfaz Proceso

Proceso proceso2			
Nombre:	proceso2		
Sub Proceso Actual	1		
Lista de Sub Procesos proceso2			
Nombre	Fecha de realización	Estado	
Task 2 of proceso2	09/05/2017 06:42:36	No disponible	ρ
Task 1 of proceso2	05/04/2017 06:42:36	Disponible	ρ
Volver			

Figura 18 Pantalla de Proceso

Detalles del proceso	
Name:	Tarea 1 de hgcc
Número en proceso	1
Fecha	31/07/2017 08:50:09
Disponibilidad	3 días a partir de la fecha
Estado	Disponible
Subir Fotografías	

Detalles del proceso	
Name:	Tarea 1 de filtroblue
Número en proceso	1
Fecha	22/05/2017 11:25:56
Disponibilidad	3 días a partir de la fecha
Estado	Finalizado

Figura 19 Información de un proceso



Figura 20 Pantalla subir fotografías

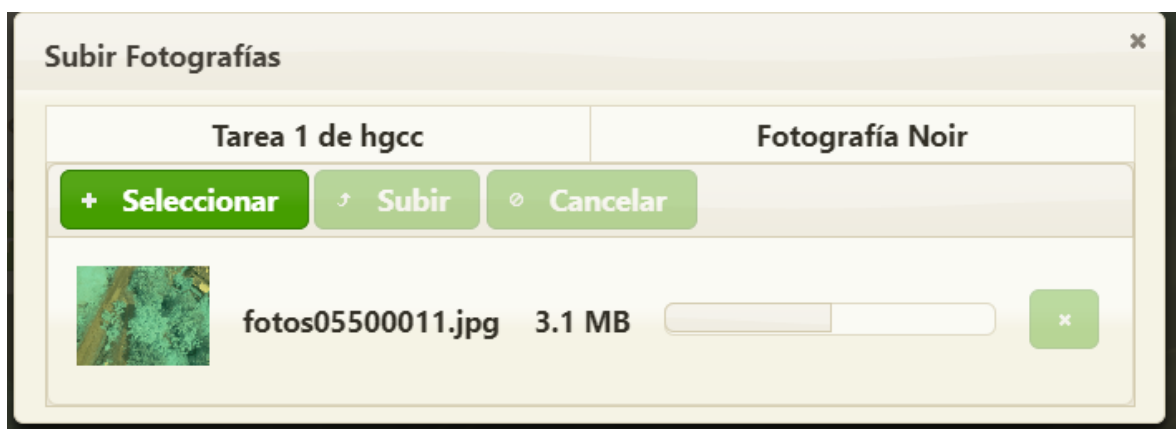


Figura 21 Pantalla subiendo fotografías

C2.4. Interfaz Subir Fotografías



Figura 22 Pantalla subir fotografías deshabilitada



Figura 23 Pantalla Subir Fotografías

Proceso filtroblue			
Nombre:	filtroblue		
Sub Proceso Actual	3		
Lista de Sub Procesos filtroblue			
Nombre	Fecha de realización	Estado	
Tarea 1 de filtroblue	22/05/2017 11:25:56	Finalizado	ρ
Tarea 2 de filtroblue	22/05/2017 11:25:56	Finalizado	ρ
Tarea 3 de filtroblue	22/05/2017 11:25:56	Finalizado	ρ
Volver Mostrar Resultados			

Figura 24 Pantalla proceso mostrar resultados

Interpretación de resultados



Plantas Sanas y vigorosas



Plantas enfermas o suelo

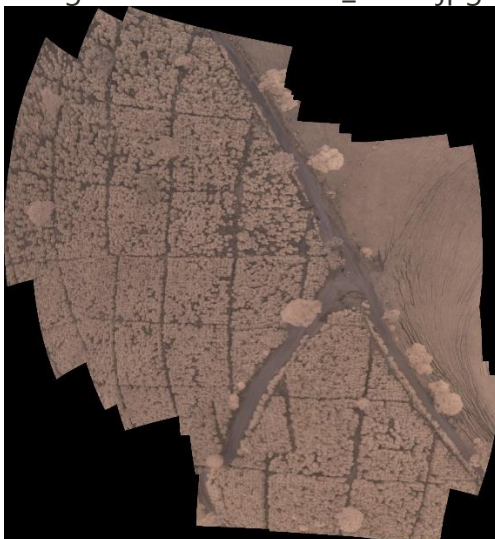


Agua u otros

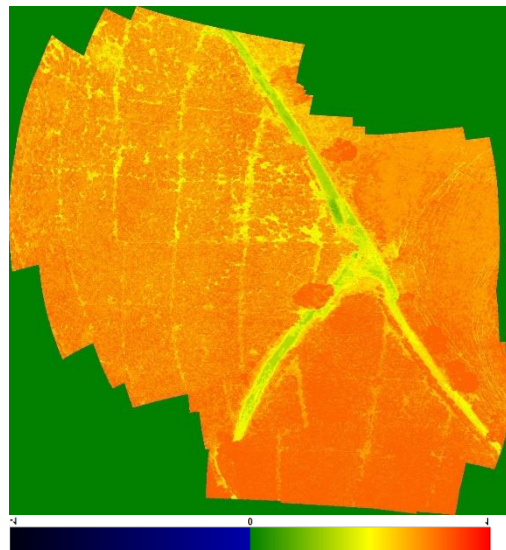


Resultados Proceso Tarea 1 de filtroblue

gan1filblue1100021_stitch.jpg



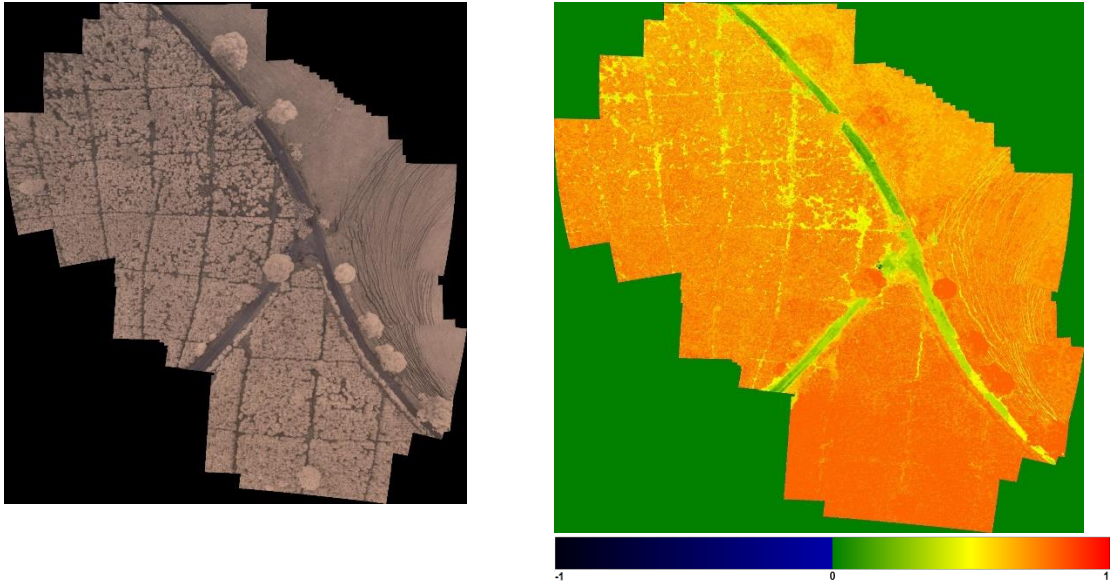
Resultado NDVI



Resultados Proceso Tarea 2 de filtroblue

Imagen fotomayo7filrojotimbiofilblue
1100039_stitch.jpg

Resultado NDVI



Resultados Proceso Tarea 3 de filtroblue

mayocatorblue1100025_stitch.jpg Resultado NDVI

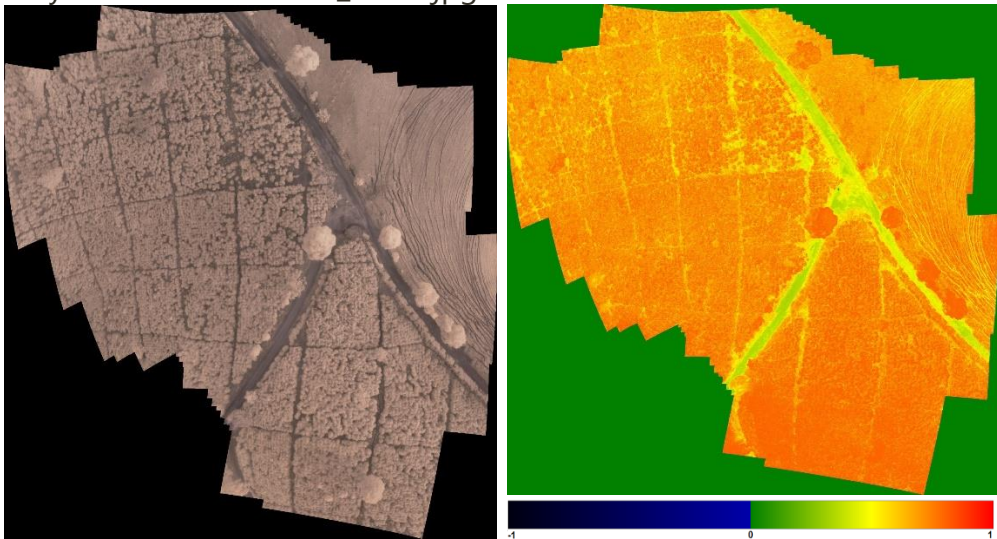


Figura 25 Ejemplo de ventana resultados AppWeb

Anexo D

Implementación Prototipo

D1. Aplicación Web

D1.1. Managed Beans o Beans Administrados.

Los Managed Beans son los componentes más importantes de la aplicación y se encargan de controlar la ejecución y el desarrollo de los procesos, así como también de la navegación entre interfaces [24]. Es posible agregar tantos Managed Bean como se desee, pues a diferencia de lo analizado en la plataforma del prototipo móvil, donde es necesario una actividad para cada interfaz, en este caso, es posible agregar uno o más ManagedBean según el requerimiento, para este caso se decide implementar 3 Managed Bean, el primero se denomina LoginBean y será el encargado de manejar toda la información de los usuarios registrados y los procesos de control de sesión; el segundo se denomina MainMB y se le llama principal por que se encarga de manejar toda la información de los procesos y sus tareas; el tercero se denomina ArchivesMB y es el encargado de manejar la información y el control del proceso de subir imágenes al servidor. A continuación se presenta las clases con las cuales se realiza la implementación de los Managed Bean.

MainMB	LoginBean	ArchivesMB
<ul style="list-style-type: none"> - LoginBean logBean - ProcessEJB processEjb - SubProcessEJB subProcessEjb - List<Proceso> processTable - Proceso processSelect - List<SubProceso> subProcessTable - SubProceso subProcessSelect - boolean subirFotos - boolean selectypephoto 	<ul style="list-style-type: none"> - UserEJB userEjb - String username - String password - String idUsuario - Usuario usu - boolean login - UploadedFile file - boolean uploadFileState 	<ul style="list-style-type: none"> - String destination - String destinationImage - LoginBean logBean - MainMB mainmb - File prueba - StreamedContent imageen - SubProcessEJB ejbSubProcess - ProcessEJB ejbProcess - boolean stateNgb - boolean stateRgb - boolean subirFotos - boolean selectypephoto
<ul style="list-style-type: none"> + MainMB() + void init() + boolean isSelectypephoto() + void setSelectypephoto(boolean selectypephoto) + boolean isSubirFotos() + void setSubirFotos(boolean subirFotos) + LoginBean getLogBean() + void setLogBean(LoginBean logBean) + List<SubProceso> getSubProcessTable() + void setSubProcessTable(List<SubProceso> subProcessTable) + SubProceso getSubProcessSelect() + void setSubProcessSelect(SubProceso subProcessSelect) + List<Proceso> getProcessTable() + void setProcessTable(List<Proceso> processTable) + Proceso getProcessSelect() + void setProcessSelect(Proceso processSelect) + void seleccionarSubProceso() + void seleccionarTipoFoto() + void verFotoNoir() + void verFotoRGB() + String abrirProceso() + String backpage() + String backpageUpload() 	<ul style="list-style-type: none"> + Usuario getUsu() + void setUsu(Usuario usu) + String getIdUsuario() + void setIdUsuario(String idUsuario) + boolean isLogin() + void setLogin(boolean login) + String getPassword() + void setPassword(String password) + boolean estaLogeado() + String getUsername() + void setUsername(String username) + String login() + String logout() + void invalidarSession() + String iniciarSistema() + UploadedFile getFile() + void setFile(UploadedFile file) + void handleFileUpload(FileUploadEvent event) + StreamedContent prepDownload() + boolean isUploadFileState() + void setUploadFileState(boolean uploadFileState) 	<ul style="list-style-type: none"> + void init() + String uploadPhotos() + boolean isSelectypephoto() + void setSelectypephoto(boolean selectypephoto) + boolean isSubirFotos() + void setSubirFotos(boolean subirFotos) + StreamedContent getImageen() + void setImageen(StreamedContent imageen) + File getPrueba() + void setPrueba(File prueba) + String getDestination() + void setDestination(String destination) + String getDestinationImage() + void setDestinationImage(String destinationImage) + boolean isStateRgb() + void setStateRgb(boolean stateRgb) + boolean isStateNgb() + void setStateNgb(boolean stateNgb) + LoginBean getLogBean() + void setLogBean(LoginBean logBean) + MainMB getMainmb() + void setMainmb(MainMB mainmb) + void handleFileUploadNoir(FileUploadEvent event) + void handleFileUploadRGB(FileUploadEvent event) + void deleteFileNoir() + void deleteFileRGB() + void copyFile(String fileName, InputStream in, String tipo) + void deleteFile(String fileName, String tipo) + void borrarDirectorio(File directorio) + void irUploadPhotoNoir() + void irUploadPhotoRGB() + void verFotoNoir() + void verFotoRGB() + String finalizarSubProceso()

Figura 26 Clases Beans Administrados

En la Figura 26 se muestran las clases más importantes del prototipo Web, que son los Beans Administrados, estas se encargan del manejo de la navegación y de organizar la lógica de los procesos del servidor Web.

D1.2. Conexión a Bases de Datos.

Esta sección se compone de diferentes elementos entre los que se encuentran los Beans de Java Empresarial (EJB) y los Handlers, los cuales se encargarán de facilitar las labores de conexión a la base de datos.

D1.2.1. Enterprise JavaBeans (EJB):

Son utilizados para encapsular toda la conexión a la base de datos con los métodos que se requieran, la ventaja más importante de utilizar estos elementos es que no es necesario que sean declarados ni creados solo se requiere instanciarlos. A lo largo de este trabajo de grado se tienen 3 EJB que serán descritos a continuación,

- **ProcessEJB:**

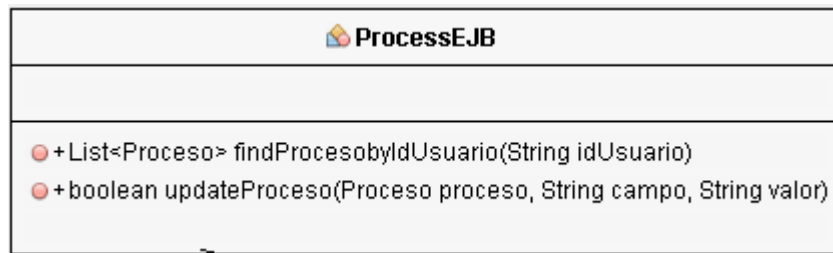


Figura 27 EJB Proceso.

La Figura 27 muestra que este EJB se encarga de listar todos los procesos por Id de usuario además del método de actualizar proceso. Cada método se encarga de crear la conexión utilizar lo necesario y luego cerrarla para evitar problemas de seguridad.

- **UserEJB:**

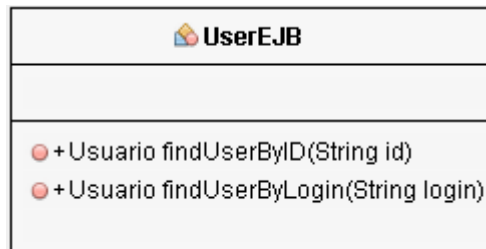


Figura 28 EJB Usuario

La Figura 28 muestra que el EJB Usuario se compone de dos métodos que se encargan de buscar los usuarios por los parámetros de ID y login o nombre de usuario.

- **SubProcessEJB:**

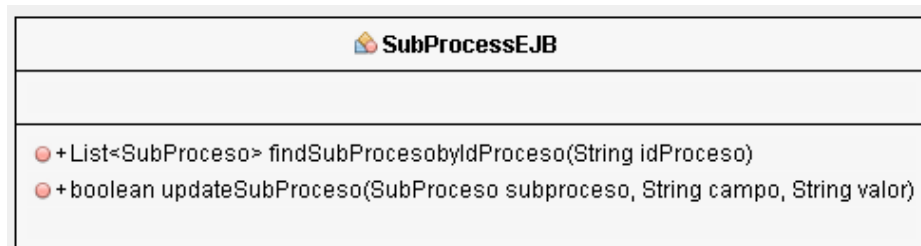


Figura 29 EJB Sub Proceso

La Figura 29 muestra que el EJB Sub Proceso se compone de dos métodos, el primero lista todos los procesos por ID y el segundo se encarga de actualizar la información de un sub proceso determinado.

D1.2.2. Handlers:

Son los encargados de llevar a cabo una comunicación directa con la base de datos mediante el *driver* proporcionado por el gestor de la base de datos, en este caso como el Servicio móvil montado para la aplicación móvil maneja *SQL Server* se debe utilizar la lógica y lenguaje SQL necesario para el mismo. Esto se puede encontrar en la documentación de *SQLServer* [25]. Los *handler* son un complemento de los EJB y llevan una interacción directa con la base de datos, por cada EJB se implementa un Handler, estos serán descritos a continuación.

- **HandlerProcess**

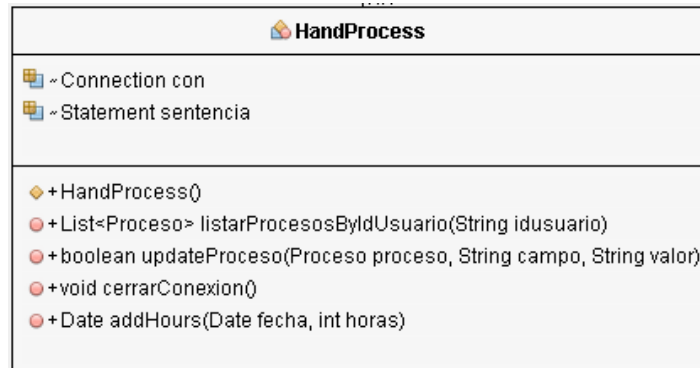


Figura 30 Clase Handler Proceso

En la Figura 30 se presenta que el *Handler* proceso contiene métodos encargados de realizar las tareas que requieran los EJB además de implementar métodos para la gestión de la conexión con el servidor.

- **HandSubProcess**

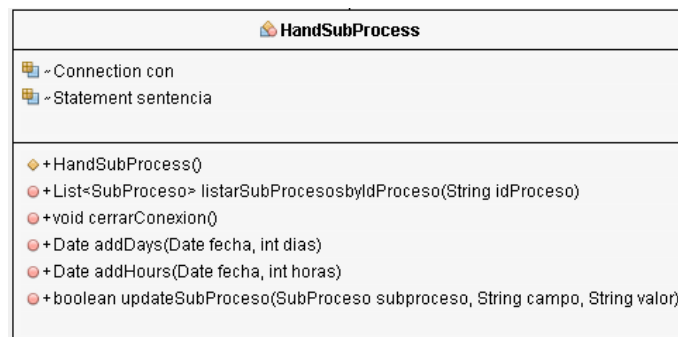


Figura 31 Handler Sub Proceso

En la clase de la Figura 31 se listan los métodos que se encargan de realizar las consultas que requiere el EJB Sub Proceso además de la gestión del as conexiones.

- **HandUser**

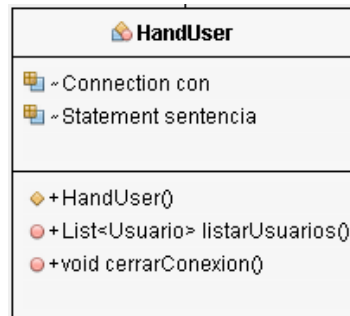


Figura 32 Handler Usuario

En la clase de la Figura 32 se listan los métodos que se encargan de implementar los métodos que requiere el EJB Usuario, además de la gestión de las conexiones.

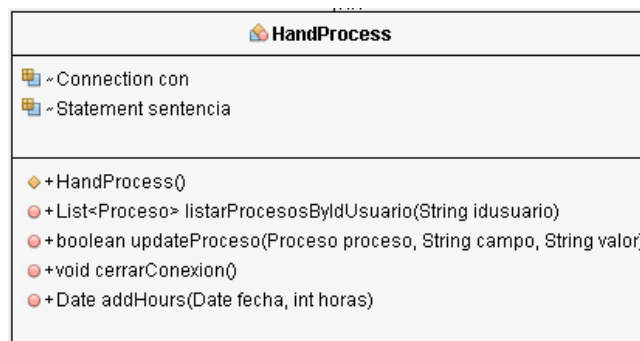


Figura 33 Implementación Handler Proceso

D2. Aplicación Móvil

D2.1. Modelos

Para comenzar con el proceso de implementación de la lógica es necesario crear los modelos de los objetos que interactúan con el sistema, estos son:

Usuario: Es la clase que mapea la información de los usuarios, en este caso los agricultores, entre sus atributos están el nombre, el correo y la contraseña.

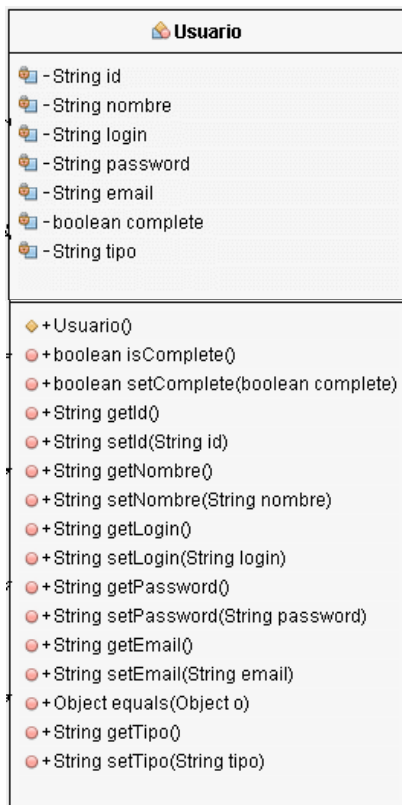


Figura 34 Clase Usuario

Proceso: Es la clase que mapea la información de los procesos que pueda crear el usuario, entre sus atributos están el estado, la fecha de inicio y de finalización, el número de pruebas entre otros.



Figura 35 Clase Proceso

SubProceso: Es la clase que mapea la información de cada una de las tareas de captura que se llevan a cabo en un proceso, entre sus características más importantes están la fecha de realización y su estado.

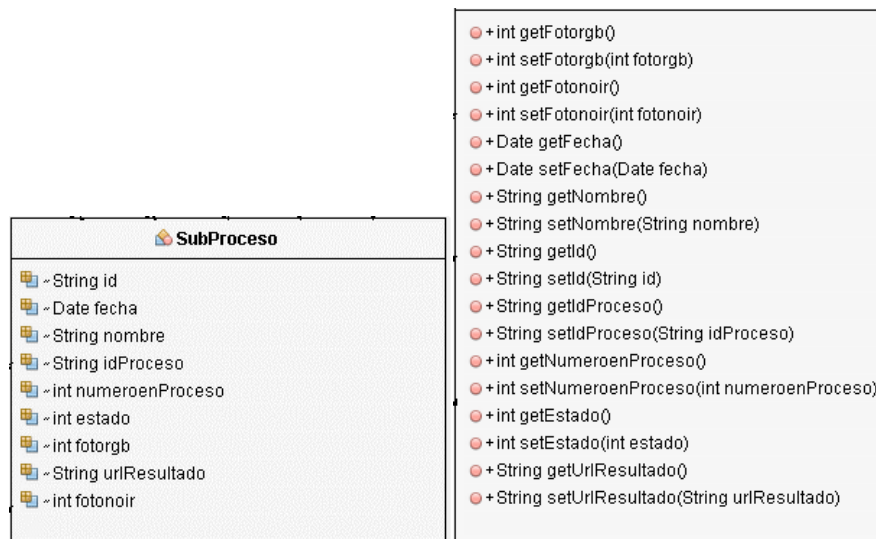


Figura 36 Clase Sub Proceso

Camara: Esta clase mapea la información de las cámaras a utilizar y sus características, entre estas están la longitud focal, la velocidad máxima de captura entre otras.

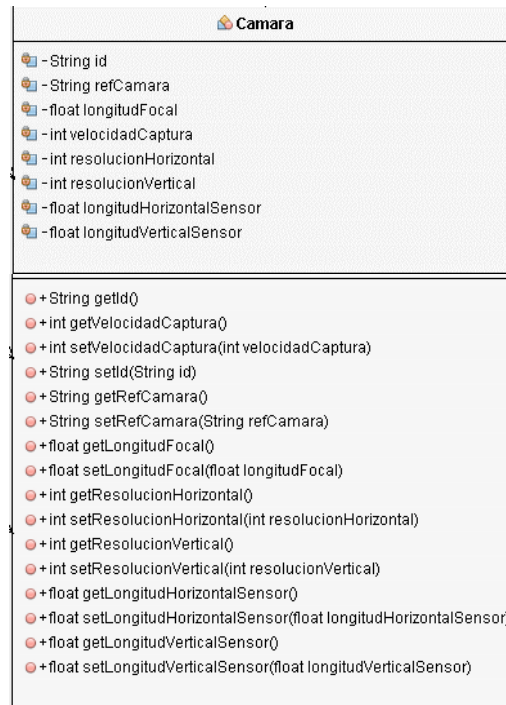


Figura 37 Clase Cámara

Aeronave: Esta clase contiene información de las aeronaves utilizadas, contiene la altura máxima, la velocidad crucero entre otras.

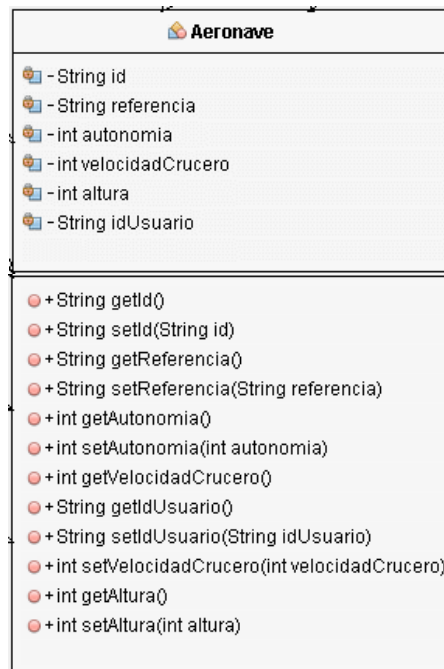


Figura 38 Clase Aeronave

D2.2. Activitys o actividades

Como se mencionaba anteriormente las actividades de la aplicación son el componente más importante de la misma, estas se encargan de renderizar la interfaz gráfica además de llevar el control de los procesos que se lleven a cabo. A continuación, se da una breve descripción de cada una de ellas.

MainActivity: La actividad principal es la que se lanza al iniciar la aplicación y se encarga de controlar los procesos de monitoreo que el usuario pueda tener.

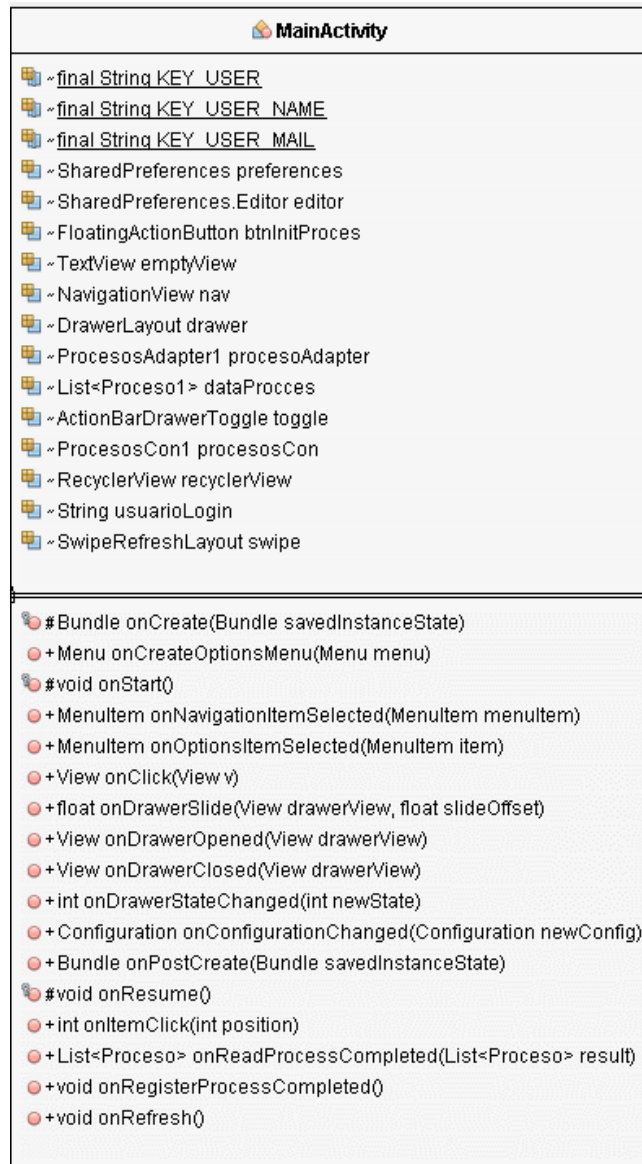


Figura 39 Clase Main Activity

NewProcessActivity: Esta actividad es la que se encarga de controlar el ingreso de nuevos procesos de monitoreo.

The screenshot displays the class hierarchy and method list for the `NewProcessActivity` class. The class hierarchy includes various UI components and data structures, while the method list includes lifecycle methods, menu handling, and business logic methods.

Class Hierarchy:

- SubProcesosCon1 subProcesosCon
- ProcesosCon1 procesosCon
- CamarasCon1 camarasCon
- View viewPrevio
- RadioGroup rdgGrupo
- int recurrency
- EditText nombre
- EditText duracionSemanas
- Button btnSave
- int contadoSubProcesosRegistrados
- SharedPreferences preferences
- SharedPreferences.Editor editor
- String usLog
- Proceso1 proceso
- List<Aeronave1> aeronaves
- List<Camara1> camaras
- ListView listViewAeronaves
- AeronavesCon1 aeronavesCon
- AircraftAdapter1 aircraftAdapter
- CameraAdapter1 cameraAdapter
- SubProceso1 subProceso1
- SubProceso1 subProceso2
- SubProceso1 subProceso3
- SubProceso1 subProceso4
- TextView headerView
- LinearLayout linear1
- RelativeLayout linear2
- int estado
- FloatingActionButton fabCameras
- Aeronave1 aeronaveSeleccionada
- Camara1 camaraSeleccionada
- TextView btAltura

Method List:

- # Bundle onCreate(Bundle savedInstanceState)
- + MenuItem onOptionsItemSelected(MenuItem item)
- + ContextMenu.ContextMenuInfo onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo)
- + MenuItem.onContextItemSelected(MenuItem item)
- + PersistableBundle onSaveInstanceState(Bundle outState, PersistableBundle outPersistentState)
- # void onDestroy()
- # void onResume()
- + int onCheckedChanged(RadioGroup group, int checkedId)
- + View onClick(View v)
- + int addDays(Date fecha, int dias)
- + List<Proceso> onReadProcessCompleted(List<Proceso> result)
- + void onRegisterProcessCompleted()
- + List<SubProceso> onReadSubProcessCompleted(List<SubProceso> result)
- + void onRegisterSubProcessCompleted()
- + List<Aeronave> onReadAircraftCompleted(List<Aeronave> result)
- + void onRegisterAircraftCompleted()
- + void onUpdateAircraftCompleted()
- + void onDeleteAircraftCompleted()
- + List<Camara> onReadCameraCompleted(List<Camara> result)
- + void onRegisterCameraCompleted()
- + void onUpdateCameraCompleted()
- + void onDeleteCameraCompleted()
- + long onItemClick(AdapterView<?> parent, View view, int position, long id)
- SubProceso addEvent(String m_selectedCalendarId, SubProceso subProceso)

Figura 40 Clase Actividad Nuevo Proceso

ProcessActivity: Esta actividad muestra los detalles de cada proceso.

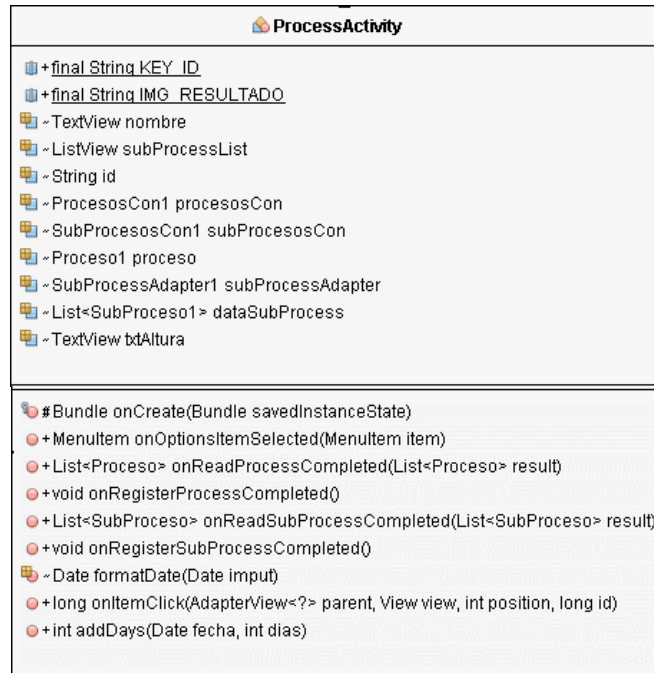


Figura 41 Clase Actividad Proceso

CamerasActivity: Esta actividad lista todas las cámaras disponibles agregadas por todos los usuarios.

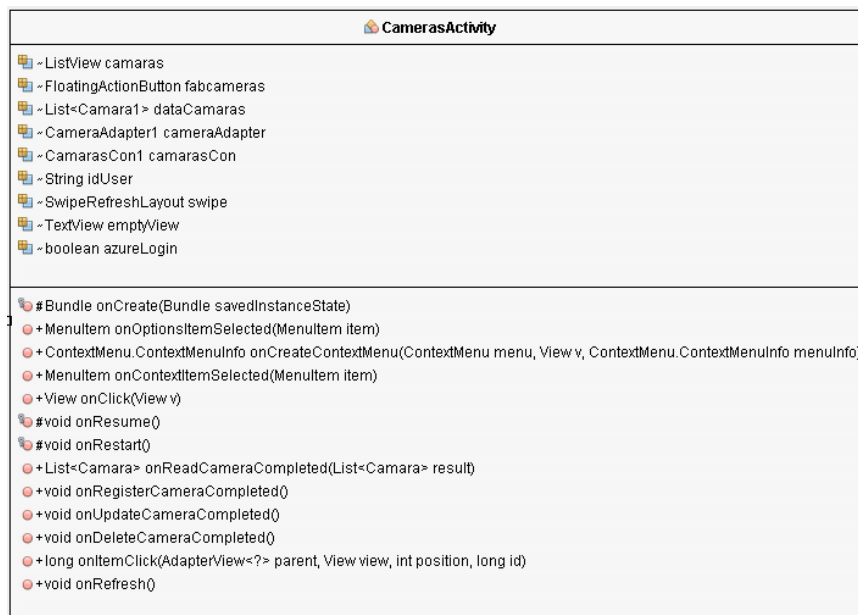


Figura 42 Actividad Cámaras

AddCameraActivity: Esta actividad permite agregar cámaras al sistema.



Figura 43 Clase Actividad Añadir Cámara

AeronavesActivity: Esta actividad lista todas las aeronaves del usuario.

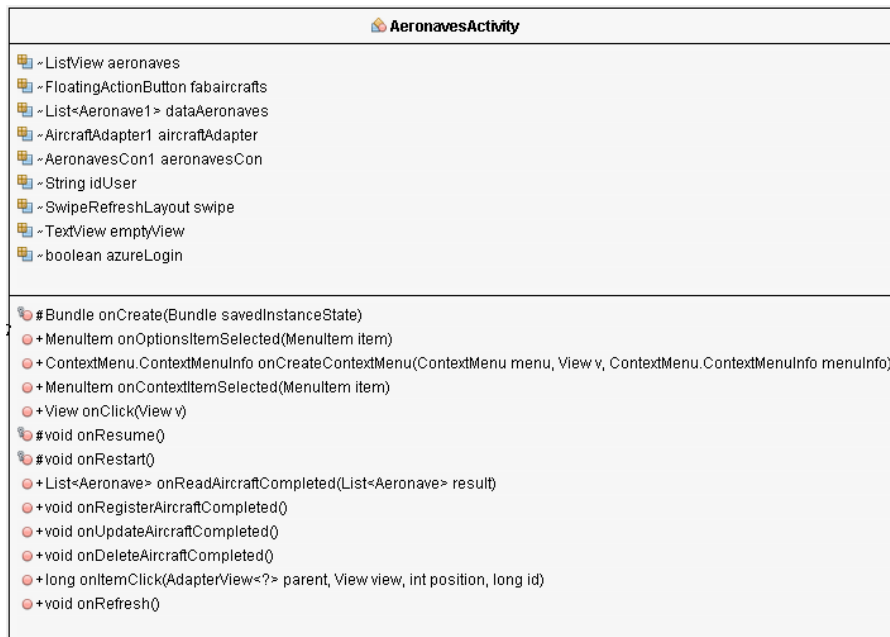


Figura 44 Actividad Aeronaves

AddUAVActivity: Esta actividad permite agregar nuevas aeronaves al sistema.

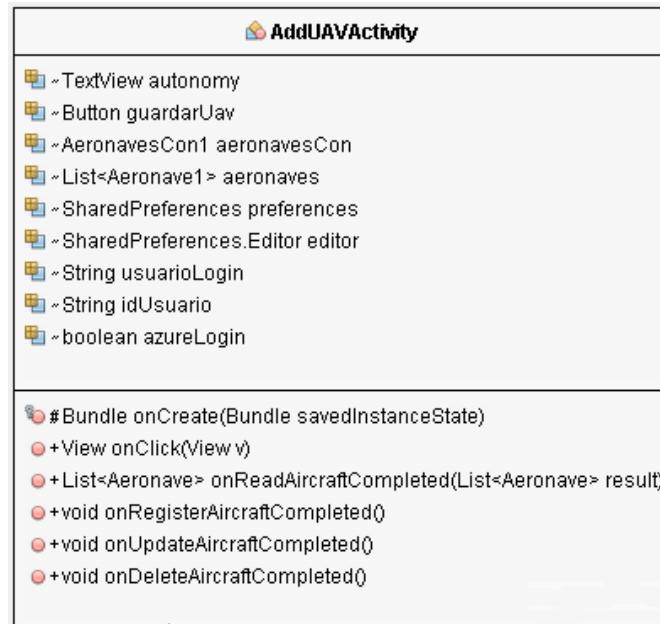


Figura 45 Clase Actividad Añadir Aeronave

LoginActivity: Esta actividad permite ingresar las credenciales para iniciar sesión.

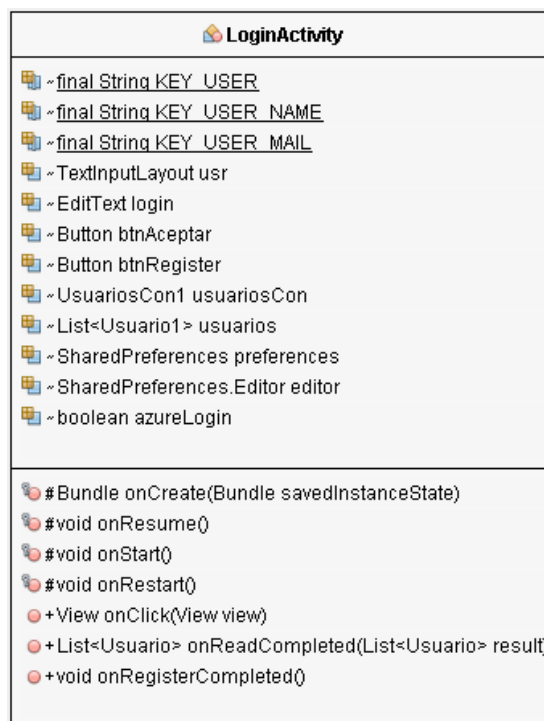


Figura 46 Clase Actividad Login

RegisterActivity: Esta actividad permite desplegar un formulario de registro para ingresar al sistema.

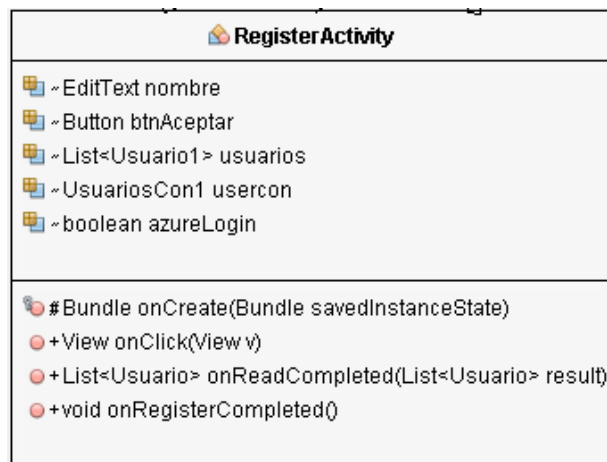


Figura 47 Clase Actividad Registro

D2.3. Adapters

Estos elementos se utilizan para llenar las listas de datos de una actividad, básicamente son útiles para aplicar una plantilla al momento de mostrar cada uno de los elementos de la lista.

AircraftAdapter: Es el adapter encargado de renderizar las aeronaves.

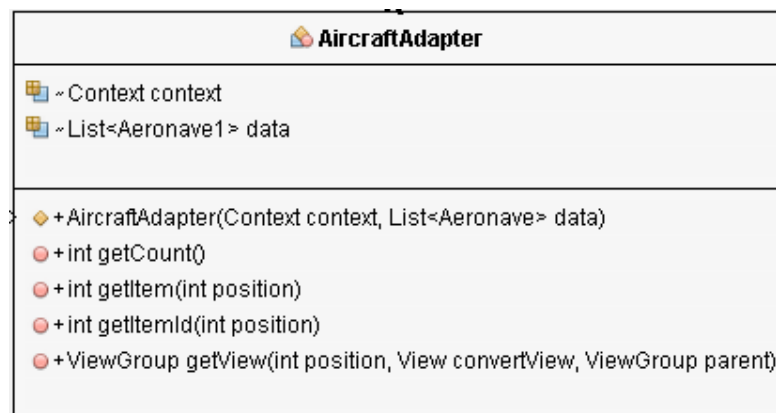


Figura 48 Clase Adapter Aeronave

ProcessAdapter: Es el encargado de renderizar los procesos de captura.

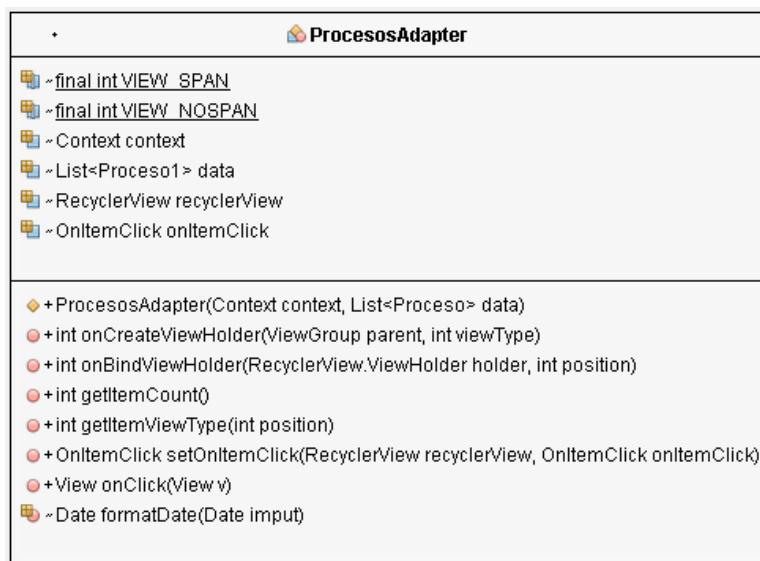


Figura 49 Clase Adapter Procesos

SubProcessAdapter: Es el encargado de renderizar los subprocesos.

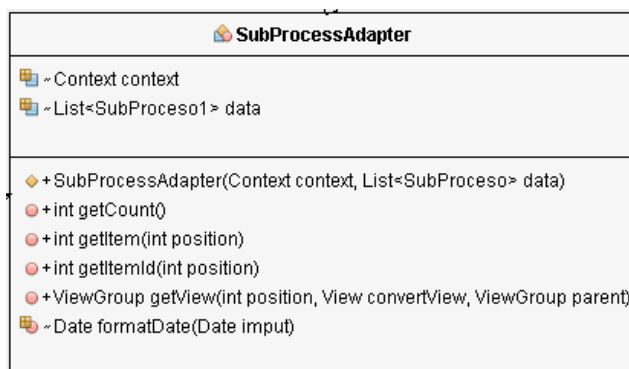


Figura 50 Clase Adapter Sub Proceso

CameraAdapter: Es el encargado de renderizar las cámaras.

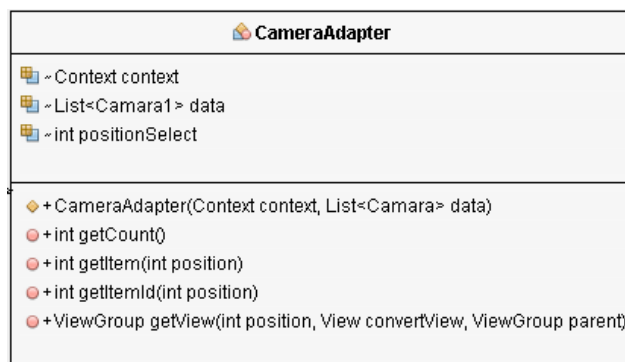


Figura 51 Clase Adapter Cámara

Como se mencionó en la sección de vistas, existen diferentes plantillas que serán utilizadas por uno o varios adpters que son los encargados de llenar las listas de datos.

Anexo E

Diagramas Prototipo

E1. Vista lógica:

En esta vista se presenta el diagrama de clases y el diagrama de secuencia del proceso más importante llevado a cabo en la aplicación.

E1.1. Diagrama de clases

El diagrama de clases que compone la vista lógica está integrado por el diagrama de clases pertenecientes a la aplicación móvil, a la aplicación web y por los modelos que intervienen, en primer caso, la aplicación móvil está compuesta por las actividades que se encargan de todo el manejo de la aplicación, se tienen actividades para controlar la gestión de usuarios y procesos; en el caso de la aplicación web se tienen los Beans Administrados que se encargan de todo el comportamiento del servidor Web, y por último se tienen los modelos, que son aquellas entidades que intervienen en ambos sistemas.

Las clases **MainMB** y **MainActivity** son las clases principales y son las que se encargan de la gestión de los procesos, como el de listarlos y a partir de ahí desplegar la información que corresponda.

La entidad **Usuario** corresponde a la representación de los usuarios a los cuales se dirige el prototipo.

La entidad **SubProceso** corresponde a una tarea definida en cada proceso y se basa en un proceso de captura en un día y hora determinado.

La entidad **Proceso** corresponde a la realización de un proceso sobre un cultivo determinado bajo unas condiciones específicas.

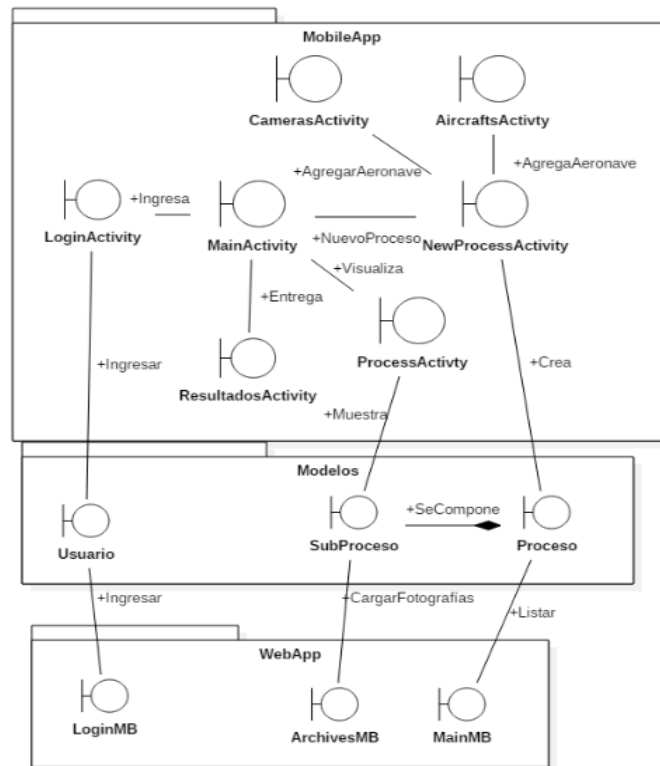


Figura 52 Diagrama de Clases

E1.2. Diagramas de secuencia

Con estos diagramas se busca describir el comportamiento del proceso más importante llevado a cabo por el prototipo que es el de la gestión de los procesos, para efectos prácticos y de entendimiento se han definido como los componentes a la aplicación móvil, la aplicación web, el servidor y la base de datos. Principalmente el usuario deberá crear un nuevo proceso ingresando la información mediante la aplicación móvil, que programa las pruebas a realizar y guarda la información en la base de datos, esta responde con un mensaje de éxito dando como resultado que el proceso ha sido creado. De forma asíncrona, la aplicación móvil verifica la fecha de las pruebas programadas y cuando esta llegue envía una alerta al usuario para que la prueba sea realizada. El usuario obtendrá las imágenes utilizando el método propuesto en el capítulo 3 y las configuraciones definidas en el capítulo 4 para subir una imagen al servidor. De esta forma el servidor procesará la imagen y al finalizar todas las pruebas el servidor entregará un resultado al usuario.

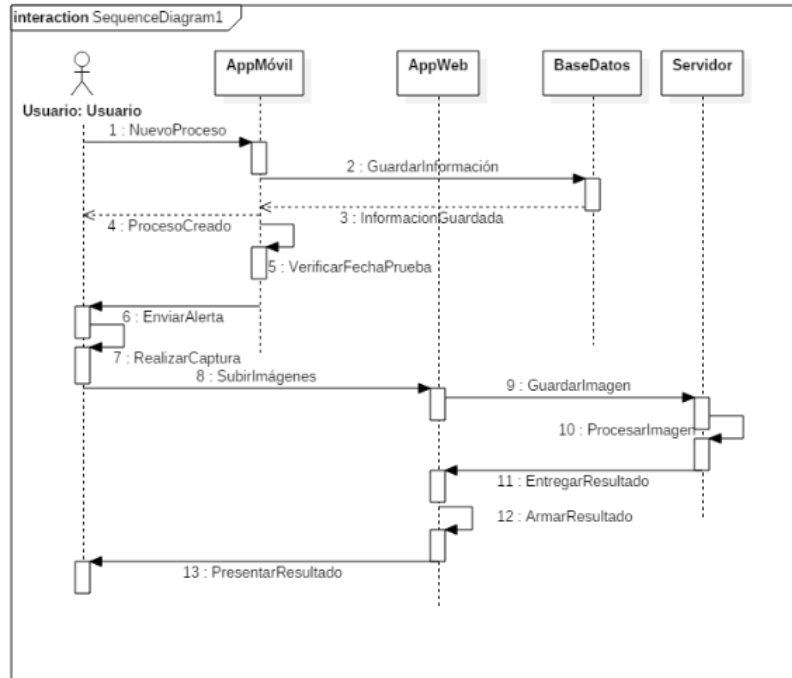


Figura 53 Diagrama de secuencia

E2. Vista de desarrollo:

Esta vista está descrita por los diagramas de componentes y paquetes.

E2.1. Diagrama de Componentes

Este diagrama indica todo los componentes que componen el sistema, en este caso se combina la información de la aplicación móvil y la aplicación web.

- **Actividades:** Este componente es el que se encarga de llevar el control de la aplicación móvil y se comunica con la base de datos.
- **AzureDriver:** Este componente se encarga de posibilitar la comunicación con la base de datos definida en la nube bajo la tecnología de Windows Azure.
- **ConexionDataBase:** Este componente se encarga de utilizar el driver de Azure y comunicarse con la base de datos, se encarga de construir las operaciones CRUD necesarias.
- **Modelos:** Los modelos son las plantillas de las entidades principales que contiene el sistema, tales como el usuario, los procesos o sub procesos.
- **Managed Beans:** Este componente se encarga del manejo de la aplicación móvil, este interactúa con los modelos y con la base de datos para el manejo de información.
- **EnterpriseJavaBeans:** Este componente se encarga de posibilitar el manejo de las conexiones a la base de datos.

- **Handlers:** Este componente se encarga de comunicarse directamente con la base de datos utilizando el Driver de JDBC para SQLServer.
- **Servidor:** Este Componente se encarga de procesar la información que se ingresa en el sistema, también entrega resultados.

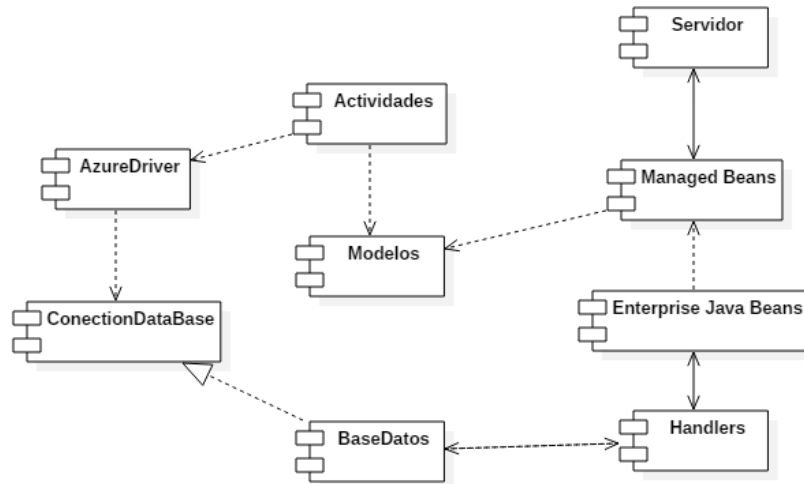


Figura 54 Diagrama de Componentes

E2.2. Diagrama de Paquetes

En la Figura 55 se puede apreciar que el diagrama de paquetes indica la organización de las diferentes clases que se encuentran en la aplicación, En el paquete **actividades** se encuentran todas las actividades que describen el sistema, en **adapters** se encuentran todas las clases encargadas de las listas que hay en la aplicación móvil, en **Net** se encuentran todas las clases encargadas de la comunicación con las bases de datos, en **Models** se encuentran todas las clases que componen las entidades principales del sistema, en **Handlers** se encuentran las clases encargadas de manejar la comunicación entre la aplicación web y la base de datos, y en **Enterprise Java Beans** se encargan las clases encargadas de facilitar la comunicación y creación de operaciones CRUD.

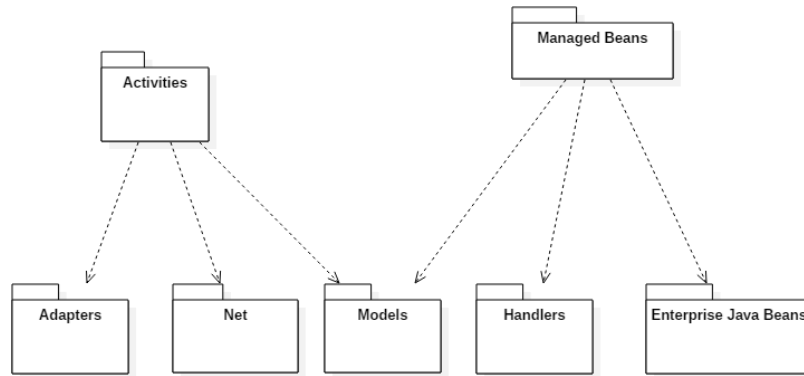


Figura 55 Diagrama de Paquetes

E3. Vista de Procesos:

Para la descripción de esta vista, es usado un diagrama de actividades, con el fin de mostrar los procesos y las formas de comunicación entre estos. En la Figura 56 se encuentra el diagrama de actividad.

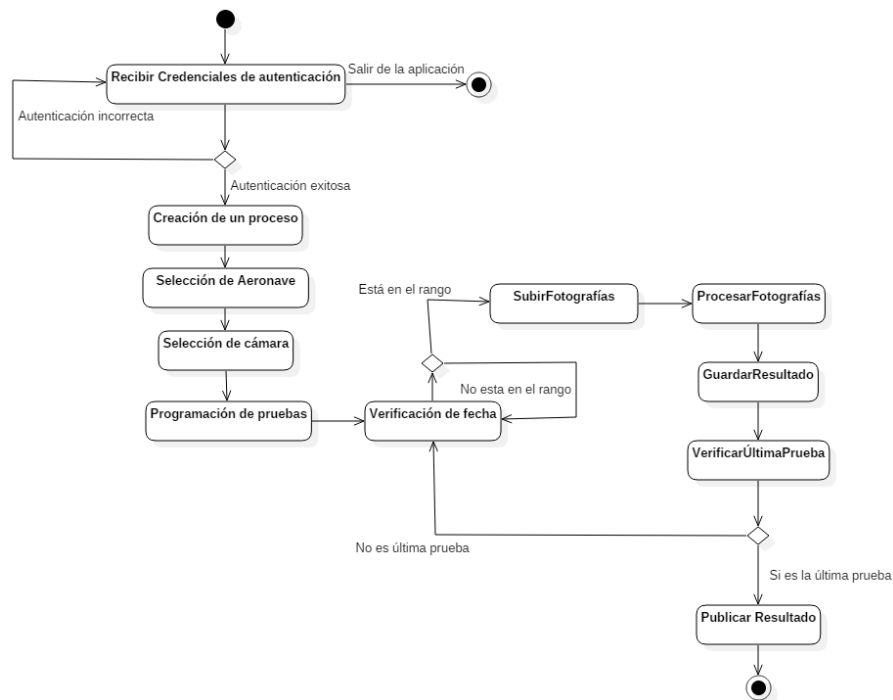


Figura 56 Diagrama de actividad

E4. Vista Física:

E4.1. Obtención de la arquitectura del sistema

La obtención de la arquitectura del sistema contempla todas las limitaciones y características de la aplicación, así como las demás vistas anteriormente descritas. En la

Figura 57 se presenta la arquitectura final del sistema. Se presenta que la aplicación móvil está principalmente compuesta por actividades, la capa de base de datos está compuesta por un servidor y por las bases de datos SQL, la aplicación web se compone principalmente de los Managed Beans y al cliente lo constituyen un navegador web y un dispositivo móvil.

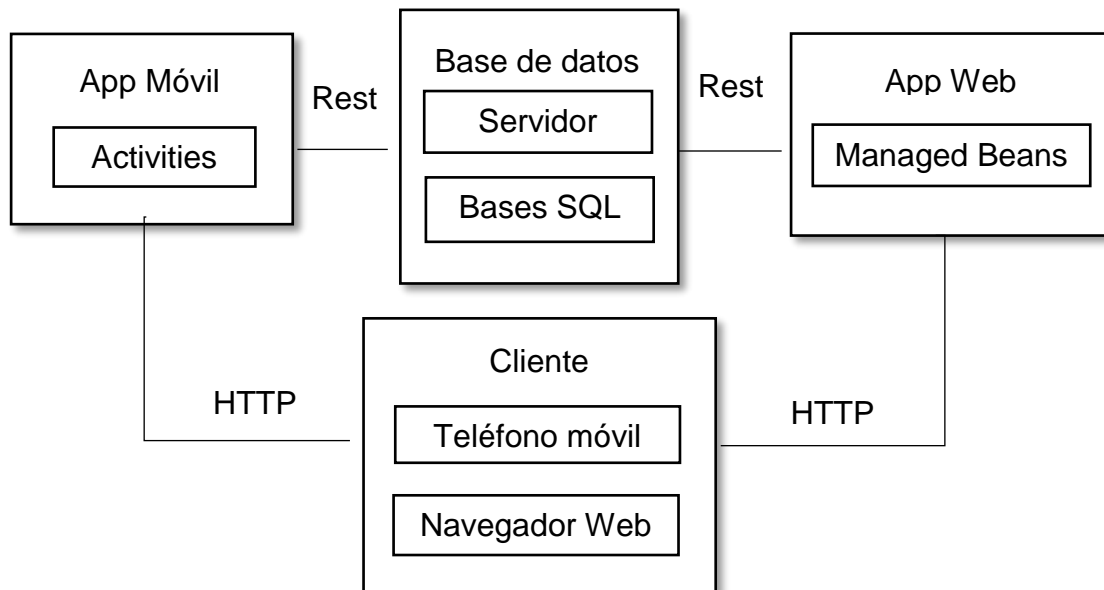


Figura 57 Arquitectura del sistema

E4.2. Diagrama de despliegue

El diagrama de despliegue de la Figura 58 muestra la agrupación de los componentes software dentro de los componentes físicos. Primeramente se encuentra que el dispositivo móvil contiene la aplicación móvil, el servidor central se encarga de desplegar la aplicación web y además de realizar el procesamiento de las imágenes, el navegador web se encarga de contener al cliente que consumirá la aplicación web, y el servidor de base de datos contiene la base de datos.

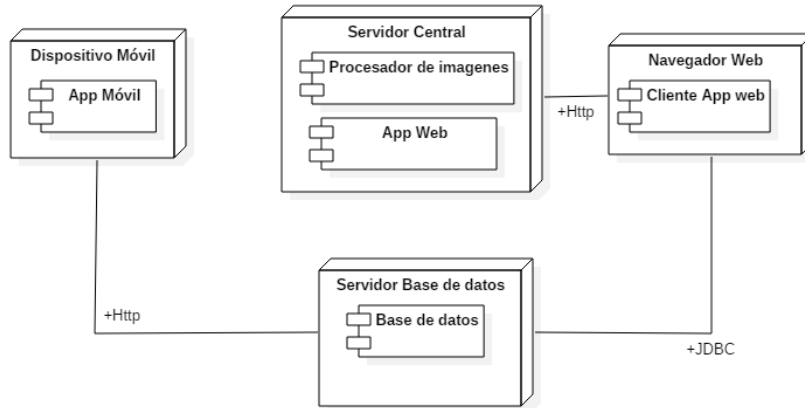


Figura 58 Diagrama de despliegue

E5. Vista de escenarios:

E5.1. Casos de uso del prototipo

En la Figura 59 se muestra el diagrama de casos de uso, quienes representan todas aquellas actividades que devuelven valor al usuario. Estos son la gestión de procesos la gestión de la sesión y la gestión de herramientas.

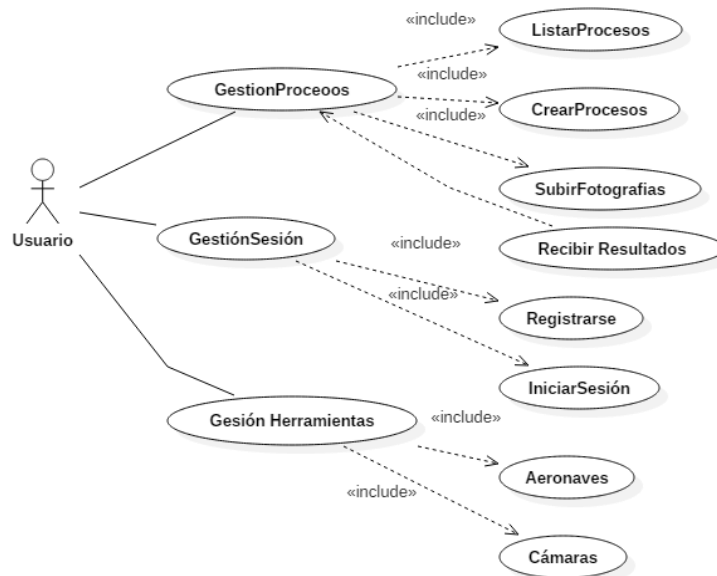


Figura 59 Diagrama de Casos de uso

A continuación se puede encontrar una descripción de cada uno de ellos utilizando las plantillas extendidas.

E5.2. Casos de Uso Detallados del Prototipo

A continuación, se describen los casos de uso del prototipo haciendo uso de plantillas, que facilitan la comprensión del diagrama de casos de uso.

Caso de uso	Listar procesos	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de listar los procesos que esté llevando a cabo el usuario.	
Precondición	Haber iniciado sesión	
Secuencia Normal	Paso	Acción
	1	El Usuario Ingresa al sistema
	2	Los Procesos son mostrados en una lista que es la lista principal de la aplicación
Postcondición	El usuario estará en la capacidad de abrir los procesos y de crear nuevos procesos.	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá listar los procesos
Importancia	Vital	

Caso de uso	Crear procesos	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de permitir a los usuarios crear un proceso de captura y procesamiento de información	
Precondición	Haber iniciado sesión	
Secuencia Normal	Paso	Acción
	1	El Usuario Ingresa al sistema
	2	Los Procesos son mostrados en una lista que es la lista principal de la aplicación
	3	El usuario selecciona la funcionalidad nuevo proceso
	4	El usuario ingresa la información del proceso
5	El usuario guarda el proceso	
Postcondición	El usuario estará en la capacidad de subir información al proceso nuevo creado en las fechas indicadas.	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá agregar nuevos procesos
Importancia	Vital	

Caso de uso	Subir Fotografías	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de permitir subir fotografías al sistema cuando el usuario esté habilitado para hacerlo	
Precondición	Haber creado un proceso y estar en las fechas establecidas por el sistema para la carga de información	
Secuencia Normal	Paso	Acción
	1	El Usuario Ingresa al sistema
	2	Los Procesos son mostrados en una lista que es la lista principal de la aplicación
	3	El usuario selecciona un proceso específico
	4	El usuario selecciona la tarea del proceso habilitada
	5	El usuario sube las fotografías al sistema
	6	El usuario da por finalizada la tarea
n		
Postcondición	El sistema habrá procesado las fotografías subidas y habrá actualizado la base de datos.	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá subir fotografías
Importancia	Vital	

Caso de uso	Recibir Resultados	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de generar los resultados para que el usuario los reciba.	
Precondición	Haber subido toda la información en las fechas estipuladas	
Secuencia Normal	Paso	Acción
	1	El Usuario Ingresa al sistema
	2	Los Procesos son mostrados en una lista que es la lista principal de la aplicación
	3	El usuario selecciona un proceso ya finalizado
	4	El usuario es capaz de generar resultados
5	El usuario exporta los resultados en PDF	
Postcondición	El usuario estará en la capacidad de abrir los procesos y de crear nuevos procesos.	
Excepciones	Paso	Acción
	1	Si no existe acceso a la red el usuario no podrá visualizar los resultados
Importancia	Vital	

Caso de uso	Registrarse	
Versión	1	
Autores	Usuario	

Descripción	Este caso de uso se encarga de permitir al usuario registrarse en el sistema.	
Precondición	Tener Conexión a internet	
Secuencia Normal	Paso	Acción
	1	El Usuario ingresa a la aplicación móvil
	2	El usuario ingresa al formulario de registro
	3	El usuario ingresa su información personal y guarda el formulario
Postcondición	El usuario estará en la capacidad de abrir los procesos y de crear nuevos procesos.	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá registrarse en el sistema
Importancia	Importante	

Caso de uso	Iniciar Sesión	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de permitir al usuario ingresar al sistema mediante la proporción de sus credenciales	
Precondición	Haberse registrado en el sistema	
Secuencia Normal	Paso	Acción
	1	El Usuario Ingresa a la aplicación
	2	El usuario proporciona sus credenciales
	3	El usuario es direccionado a la interfaz principal
Postcondición	El usuario estará en la capacidad de listar los procesos que haya iniciado	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá iniciar sesión
Importancia	Vital	

Caso de uso	Aeronaves	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de permitir al usuario la gestión de las aeronaves que intervendrán en los procesos de captura	
Precondición	Haber iniciado sesión	
Secuencia Normal	Paso	Acción
	1	El Usuario Inicia sesión
	2	El usuario ingresa al menú de aeronaves
	3	El usuario es capaz de gestionar las aeronaves
Postcondición	El usuario estará en la capacidad de incluir en sus procesos las aeronaves que tenga registradas.	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá gestionar las aeronaves

Importancia	Importante	
Caso de uso	Cámaras	
Versión	1	
Autores	Usuario	
Descripción	Este caso de uso se encarga de permitir al usuario la gestión de las cámaras que intervendrán en los procesos de captura	
Precondición	Haber iniciado sesión	
Secuencia Normal	Paso	Acción
	1	El Usuario Inicia sesión
	2	El usuario ingresa al menú cámaras
	3	El usuario es capaz de gestionar las cámaras
Postcondición	El usuario estará en capacidad de incluir en sus procesos las cámaras que estén disponibles en el sistema	
Excepciones	Paso	Acción
	1	Si existe falla en la red, el usuario no podrá gestionar las cámaras.
Importancia	Importante	

Anexo F

Anexos pruebas finales

F1. Pruebas adicionales

Adicionalmente, a las pruebas realizadas en el cultivo de timbío, a lo largo de este trabajo de grado se llevaron a cabo muestreos en diferentes cultivos los cuales aportaron información y experiencia en torno a la realización de las pruebas.

F1.1. Prueba en cultivo de arroz

Se realiza una captura en un cultivo de alrededor de 3 meses, posteriormente se repite la captura un mes después con lo cual se puede tener una perspectiva en el tiempo de la evolución del desarrollo del cultivo durante ese tiempo. Debido a inconvenientes presentados durante la captura de la segunda prueba solo fue posible realizar la prueba mediante la configuración AWB 11 filtro rojo. Los resultados se presentan a continuación:



Figura 60 Resultados de la captura Filtro Rojo AWB 11

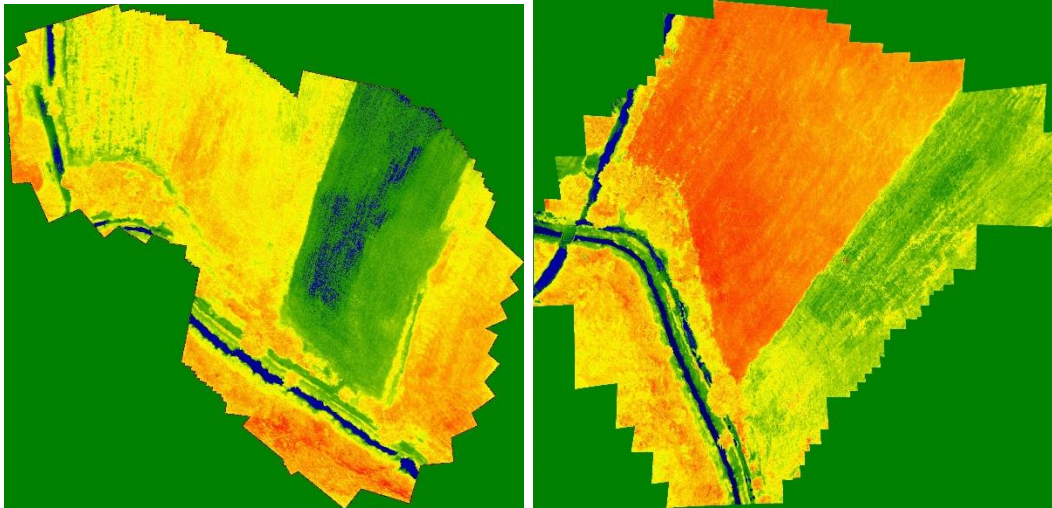


Figura 61 Resultados NDVI Captura Yaguará

En la Figura 61 se puede encontrar el resultado NDVI que resulta del procesamiento de las dos imágenes, como se puede apreciar en la imagen del lado izquierdo, que corresponde a la primera imagen capturada, el cultivo de arroz no tiene una uniformidad en torno a la vegetación, y por el contrario, la imagen de la derecha, que corresponde a una captura realizada un mes después, posee una concentración del cultivo de arroz mucho más elevada, con ello es posible si se desea tener en cuenta los lugares de mayor evolución así como también llevar un control de las zonas de mayor productividad.

F2. Mosaicos Sobrevuelos Realizados

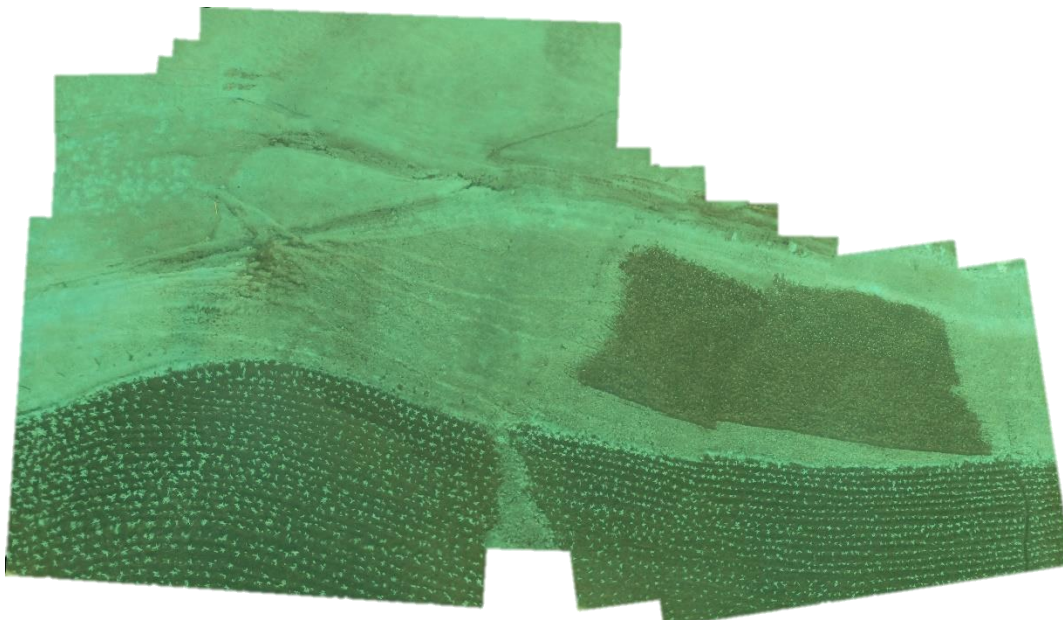


Figura 62 Sobrevuelo Villamoreno Nariño Cultivo de Arveja

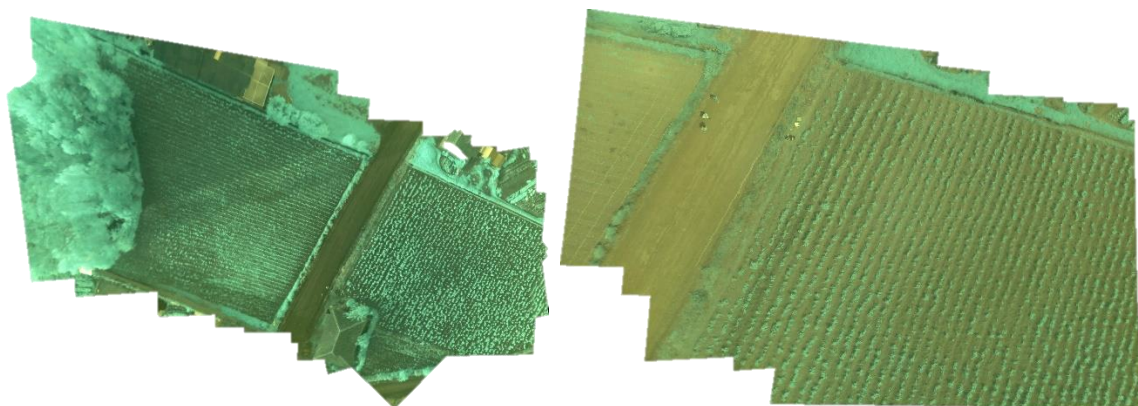


Figura 63 Sobrevuelo San Fernando Nariño Cultivo de Arveja

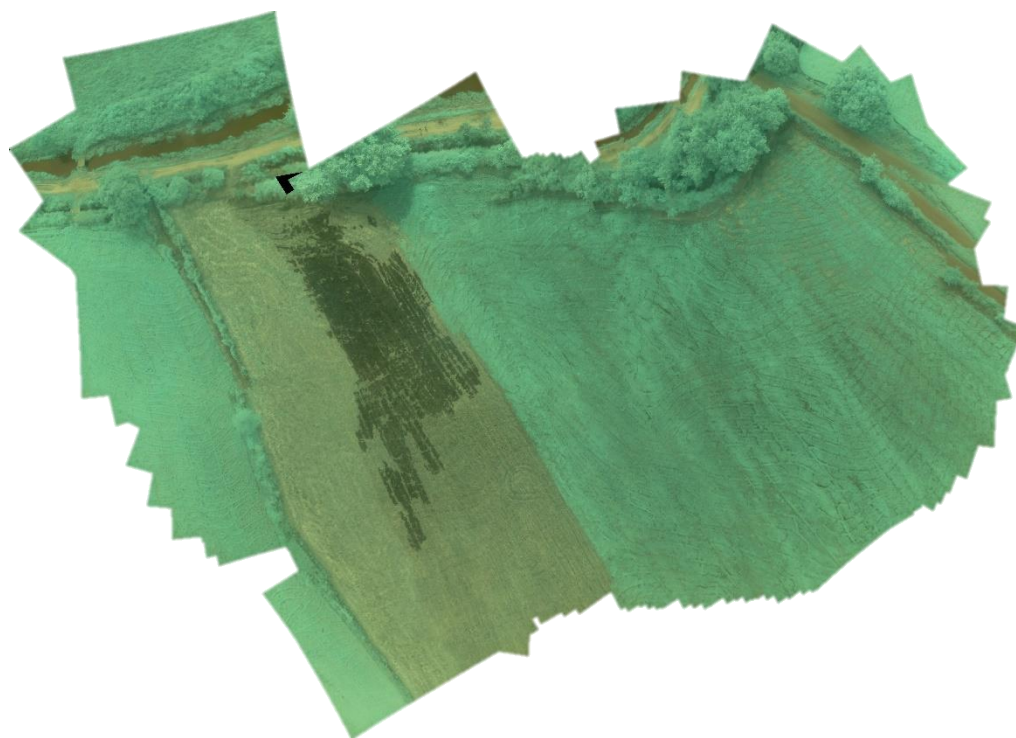


Figura 64 Sobrevuelo Yaguará Huila Cultivo de Arroz

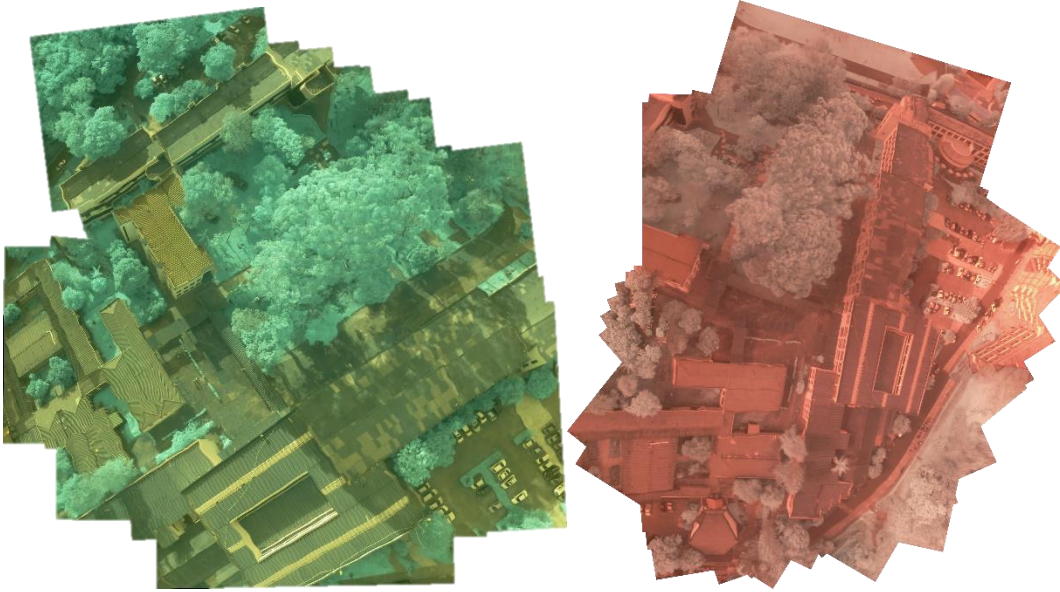


Figura 65 Sobrevuelo Campus Universitario

F3. Datos completos pruebas finales

A continuación se presenta las diferentes tablas las cuales se componen de los datos de los diferentes hilos de ejecución realizados, estos datos representan la información arrojada por el programa al realizar el proceso de clasificación de información.

F3.1. Filtro Rojo #19 con AWB 11:

Tarea 1 filtro rojo 11					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	2828539	486208	1117060	4431807	4437471
2	3324008	544385	561133	4429526	4437471
3	1221078	1163304	2050502	4434884	4437471
4	708248	259777	3467002	4435027	4437471
5	3893787	395027	144565	4433379	4437471
6	4059182	373984	0	4433166	4437471
7	2235367	1751550	450173	4437090	4437471
8	615906	3338803	479875	4434584	4437471
9	1142866	42532	3246523	4431921	4437471
10	3554280	543782	334278	4432340	4437471
11	3175915	1013961	245212	4435088	4437471
12	977541	2436625	1019459	4433625	4437471
13	421464	204316	3807960	4433740	4437471

14	3400894	153437	875242	4429573	4437471
15	3649024	348782	427086	4424892	4437471
16	310691	558718	3565162	4434571	4437471

Tabla 2 Resultados Tarea 1 Filtro rojo awb 11

tarea 2 filtro rojo 11					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	3372543	597938	764223	4734704	4757600
2	3403598	1017824	310378	4731800	4757600
3	1265322	929873	2557388	4752583	4757600
4	59982	45571	4650182	4755735	4757600
5	3405177	211528	1118621	4735326	4757600
6	4182971	539963	2	4722936	4757600
7	1685423	2587679	481974	4755076	4757600
8	745318	3040377	968334	4754029	4757600
9	386811	42734	4323842	4753387	4757600
10	3676264	561873	500642	4738779	4757600
11	3447873	949870	339225	4736968	4757600
12	1814485	2595149	344152	4753786	4757600
13	0	0	4757600	4757600	4757600
14	1778204	152719	2813724	4744647	4757600
15	4221056	213082	290713	4724851	4757600
16	1740917	647145	236095	2624157	4757600

Tabla 3 Resultados Tarea 2 Filtro rojo awb 11

Tarea 3 filtro rojo 11					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	2935902	302109	2571694	5809705	5896385
2	3059024	1156127	1606613	5821764	5896385
3	1341628	780313	3678542	5800483	5896385
4	2288	14450	5879547	5896285	5896385
5	4472001	439188	919990	5831179	5896385
6	5365017	461142	0	5826159	5896385
7	2973890	2240819	524732	5739441	5896385
8	2670921	797872	2422385	5891178	5896385
9	1317901	46836	4519850	5884587	5896385
10	4159433	217143	1508157	5884733	5896385

11	4724625	711222	253079	5688926	5896385
12	3406387	1600872	831170	5838429	5896385
13	0	0	5896385	5896385	5896385
14	172956	220093	5502528	5895577	5896385
15	4519320	244345	1092967	5856632	5896385
16	3405277	187479	2231675	5824431	5896385

Tabla 4 Resultados Tarea 3 Filtro rojo awb 11

F3.2. Filtro Rojo #19 con AWB 0.54,1:

Tarea 1 filtro rojo 0541					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	588255	11911	1180098	1780264	2872000
2	916811	206477	286041	1409329	2872000
3	2292314	186820	268247	2747381	2872000
4	1288433	63517	1518103	2870053	2872000
5	421011	1958	52602	475571	2872000
6	368216	28861	0	397077	2872000
7	868921	569811	383084	1821816	2872000
8	2574187	175059	118476	2867722	2872000
9	265099	1925	547906	814930	2872000
10	193846	88270	77612	359728	2872000
11	492500	333905	175522	1001927	2872000
12	1686725	444281	331703	2462709	2872000
13	45760	1304	2059412	2106476	2872000
14	194201	230474	753402	1178077	2872000
15	95180	6727	198604	300511	2872000
16	143812	61407	1539453	1744672	2872000

Tabla 5 Resultados Tarea 1 Filtro azul AWB 0.54,1

Prueba 4: tarea 2 filtro rojo 0541					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	381931	5726	3517450	3905107	5703192
2	473598	115926	646948	1236472	5703192
3	2557665	242364	1672180	4472209	5703192
4	179707	447	5498241	5678395	5703192
5	971459	3683	50564	1025706	5703192

6	281083	146	0	281229	5703192
7	1963806	449163	200197	2613166	5703192
8	4395637	3532	1101622	5500791	5703192
9	23740	1125	5412645	5437510	5703192
10	130373	65991	1919866	2116230	5703192
11	758355	614521	115949	1488825	5703192
12	3935576	298203	606088	4839867	5703192
13	0	0	5703192	5703192	5703192
14	44236	5872	5389732	5439840	5703192
15	253919	683	565497	820099	5703192
16	206797	33293	3466077	3706167	5703192

Tabla 6 Resultados Tarea 2 Filtro azul AWB 0.54,1

Prueba 5: tarea 3 filtro rojo 0541					
	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	1619275	130153	2185116	3934544	5725608
2	1275785	265406	61514	1602705	5725608
3	1909366	161184	3157543	5228093	5725608
4	337868	1259	5377619	5716746	5725608
5	2556179	115859	640771	3312809	5725608
6	694677	2267	0	696944	5725608
7	2349426	924503	231802	3505731	5725608
8	3527942	5871	2067961	5601774	5725608
9	327466	3666	4703031	5034163	5725608
10	489514	152327	1204506	1846347	5725608
11	853557	402198	93017	1348772	5725608
12	3732849	263887	707192	4703928	5725608
13	0	0	5725608	5725608	5725608
14	126772	25062	5045874	5197708	5725608
15	471192	793	879090	1351075	5725608
16	448860	81741	3647920	4178521	5725608

Tabla 7 Resultados Tarea 1 Filtro azul AWB 0.54,1

F3.3. Filtro Azul #2007 con AWB 11:

Tarea 1 filtro azul awb 11					
	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	2279915	24416	1617229	3921560	4672349

2	3007472	170333	622264	3800069	4672349
3	1028009	151588	3327366	4506963	4672349
4	47122	494	4615727	4663343	4672349
5	2983204	8065	86026	3077295	4672349
6	3367956	13469	0	3381425	4672349
7	2966046	428019	8969	3403034	4672349
8	2862142	1592	819314	3683048	4672349
9	1456623	2850	1920845	3380318	4672349
10	2593894	14099	0	2607993	4672349
11	1911543	330917	0	2242460	4672349
12	1695857	28142	522988	2246987	4672349
13	15329	242	4632101	4647672	4672349
14	597431	45342	2484976	3127749	4672349
15	452229	1097	347865	801191	4672349
16	716297	8008	1095871	1820176	4672349

Tabla 8 Resultados Tarea 1 Filtro azul AWB 11

Tarea 2 filtro azul					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	2082364	60564	2323304	4466232	5052670
2	3862727	466121	75251	4404099	5052670
3	2504993	111267	2130408	4746668	5052670
4	211743	1002	4832842	5045587	5052670
5	2647776	20727	547818	3216321	5052670
6	3293744	68544	2	3362290	5052670
7	3245453	653061	1620	3900134	5052670
8	2892058	23383	1272479	4187920	5052670
9	154617	2376	4711110	4868103	5052670
10	2091702	194911	764059	3050672	5052670
11	1806309	470164	767	2277240	5052670
12	1747413	121880	648918	2518211	5052670
13	0	0	5052670	5052670	5052670
14	28256	4067	4142677	4175000	5052670
15	333466	1110	438288	772864	5052670
16	557515	66879	2000521	2624915	5052670

Tabla 9 Resultados Tarea 2 Filtro azul AWB 11

Prueba 9: tarea 3 filtro azul					
Hilo	Plantas sanas	Plantas con problemas o suelo	Agua u otros	Total	Total prueba
1	2170034	6322	1384896	3561252	4251300
2	3001819	93096	229438	3324353	4251300
3	2273535	317128	982372	3573035	4251300
4	1293963	8292	2824525	4126780	4251300
5	2413637	3265	488499	2905401	4251300
6	2513537	8510	0	2522047	4251300
7	2613408	347381	34	2960823	4251300
8	2395118	52920	635897	3083935	4251300
9	582441	848	3171326	3754615	4251300
10	1672285	49714	760440	2482439	4251300
11	1726010	269876	0	1995886	4251300
12	2040646	123344	128126	2292116	4251300
13	0	0	4251300	4251300	4251300
14	246984	28695	3656883	3932562	4251300
15	539377	639	738755	1278771	4251300
16	527529	1074	1873778	2402381	4251300

Tabla 10 Resultados Tarea 2 Filtro azul AWB 11

Anexo G

Manual de configuración Prototipo

G1. Instalación aplicación móvil

La aplicación móvil está optimizada para trabajar sobre un teléfono móvil Android con un SDK superior a una versión 23, para la instalación se realizan los siguientes pasos.

Nota: Los directorios Github de código fuente están en:

Aplicación móvil: <https://goo.gl/1AtHcm>

Aplicación Web: <https://goo.gl/BPBsNo>

G1.1. Descarga APK

Existen dos versiones de la aplicación a instalar, su diferencia radica en que una permite programar procesos de caracterización para su uso inmediato y la otra permite realizar procesos de captura con una programación automática de acuerdo a lo deseado. Se encuentra el apk directamente en los siguientes enlaces:

- **Aplicación con fechas automáticas**
<http://mvubuntu16.eastus.cloudapp.azure.com/ApkMovil/AgroMobileConFechas.apk>
- **Aplicación sin fechas automáticas**
<http://mvubuntu16.eastus.cloudapp.azure.com/ApkMovil/AgroMobileSinFechas.apk>

Luego de descargar el APK se procede a instalarla con lo que finaliza el proceso. La imagen a mostrar inicialmente se presenta en la figura

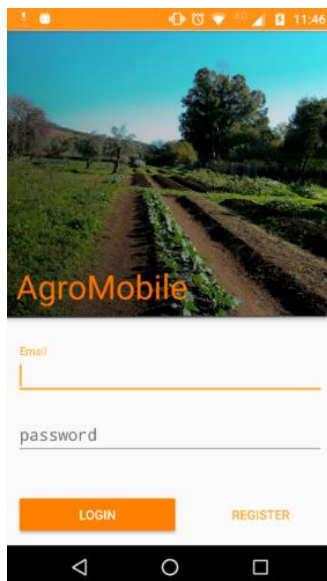


Figura 66 Vista iniciar sesión

Posteriormente a la instalación se debe realizar un registro al sistema tal como se muestra en la figura Figura 14 .

Nota: Se deben aceptar todos los permisos que sean requeridos por el sistema en tiempo real para garantizar el funcionamiento de la aplicación.

Se recomienda agregar todas las aeronaves que posea para que estén disponibles al momento de realizar los procesos de captura.

G2. Acceso al servicio Web

Para poder iniciar sesión en el servidor basta con acceder a la URL del sitio en donde se desplegará la interfaz de bienvenida de la Figura 6, para posteriormente pasar a la ventana de Login que nos permitirá acceder al sistema

La URL de acceso es:

<https://goo.gl/AZg9Qd>

Bibliografía

- [1] J. Bendig, K. Yu, H. Aasen, A. Bolten, S. Bennertz, J. Broscheit, M. Gnyp y G. Bareth, «Combining UAV-based plant height from crop surface models, visible, and near infrared vegetation indices for biomass monitoring in barley,» *International Journal of Applied Earth Observation and Geoinformation*, pp. 79-87, 2015.
- [2] E. R. Hunt, C. S. Daughtry, S. B. Mirsky y W. D. Hively, «Remote sensing with simulated unmanned aircraft imagery for precision agriculture applications,» *electd Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, pp. 4566--4571, 2014.
- [3] J. Torres-Sánchez, J. Peña, A. De Castro y F. López-Granados, «Multi-temporal mapping of the vegetation fraction in early-season wheat fields using images from UAV,» *Computers and Electronics in Agriculture*, pp. 104--113, 2014.
- [4] J. Torres-Sánchez, F. López-Granados y J. Peña, «An automatic object-based method for optimal thresholding in UAV images: Application for vegetation detection in herbaceous crops,» *Computers and Electronics in Agriculture*, pp. 43--52, 2015.
- [5] N. Otsu, «A threshold selection method from gray-level histograms,» *IEEE Trans*, p. 62--66, 1979.
- [6] W. Wei, Y. Xia, Y.-c. TIAN, X.-j. LIU, N. Jun, W.-x. CAO y Z. Yan, «Common spectral bands and optimum vegetation indices for monitoring leaf nitrogen accumulation in rice and wheat,» *Journal of Integrative Agriculture*, pp. 2001--2012, 2012.
- [7] W. Li, Z. Niu, H. Chen, D. Li, M. Wu y W. Zhao, «Remote estimation of canopy height and aboveground biomass of maize using high-resolution stereo images from a low-cost unmanned aerial vehicle system,» *Ecological Indicators*, pp. 637--648, 2016.
- [8] A. Gitelson, Y. Kaufman, R. Stark y D. Rundquist, «Novel algorithms for remote estimation of vegetation fraction,» *Remote sensing of Environment*, pp. 76--87, 2002.

- [9] M. Guijarro, G. Pajares, I. Riomoros, P. Herrera, X. Burgos-Artizzu y A. Ribeiro, «Automatic segmentation of relevant textures in agricultural images,» *Computers and Electronics in Agriculture*, pp. 75--83, 2011.
- [10] T. Murakami, M. Yui y K. Amaha, «Canopy height measurement by photogrammetric analysis of aerial images: Application to buckwheat (*Fagopyrum esculentum* Moench) lodging evaluation,» *Computers and electronics in agriculture*, pp. 70--75, 2012.
- [11] C. A. Rokhmana, «The Potential of UAV-based Remote Sensing for Supporting Precision Agriculture in Indonesia,» *Procedia Environmental Sciences*, pp. 245--253, 2015.
- [12] P. Gábrlík, «THE TECHNIQUES OF TERRAIN MAPPING AND MODELING,» The Faculty of Electrical Engineering, Czech Republic, 2014.
- [13] Anthony, D. Elbaum, S. Lorenz, A. Detweiler y Carrick, «On crop height estimation with UAVs,» *IEEE*, pp. 4805--4812, 2014.
- [14] M. Kelly, «Weed Mapping in Early-Season Maize Fields Using Object-Based Analysis of Unmanned Aerial Vehicle (UAV) Images,» *PLOS One*, 2010.
- [15] W. Kazmi, F. J. Garcia-Ruiz, J. Nielsen, J. Rasmussen y H. J. Andersen, «Detecting creeping thistle in sugar beet fields using vegetation indices,» *Computers and Electronics in Agriculture*, pp. 10--19, 2015.
- [16] S. Post, «Sign Post Survey,» [En línea]. Available: <http://signpostsurveys.ie/>. [Último acceso: 25 mayo 2016].
- [17] T. Delair, «Delair Tech,» [En línea]. Available: <http://www.delair-tech.com/>. [Último acceso: 10 mayo 2016].
- [18] Sioma, «Sioma,» [En línea]. Available: <http://siomapp.com/>. [Último acceso: 12 mayo 2016].
- [19] Java, «Conozca más sobre la tecnología Java,» [En línea]. Available: <https://www.java.com/es/about/>. [Último acceso: 3 Febrero 2016].
- [20] S. Holden, «Python,» 4 Marzo 2014. [En línea]. Available: <https://wiki.python.org/moin/BeginnersGuide/Overview>. [Último acceso: 3 Febrero 2016].
- [21] Genbeta, «NetBeans,» 9 Enero 2014. [En línea]. Available: <http://www.genbetadev.com/herramientas/netbeans-1>. [Último acceso: 3 Febrero 2016].

- [22] «Parrot,» 2017. [En línea]. Available: <https://www.parrot.com/es/drones/parrot-ardrone-20-elite-edition#técnicos>.
- [23] «DJI,» [En línea]. Available: <https://www.dji.com/es/phantom-3-standard>. [Último acceso: 2017].
- [24] Oracle, «JEE Documentation,» 2015. [En línea]. Available: <http://docs.oracle.com/javaee/7/index.html>. [Último acceso: 1 09 2016].
- [25] Microsoft, «SQL Server Technical Documentation,» [En línea]. Available: <https://docs.microsoft.com/en-us/sql/sql-server/sql-server-technical-documentation>. [Último acceso: 10 11 2016].