



Proyecto Final de Carrera Ingeniería Informática	Informe de estado del proyecto	 
---	--------------------------------	---

REALIZADO POR	FECHA	FIRMA
Francisco Yackel	14/11/17	
REVISADO POR	FECHA	FIRMA
Leandro Vignolo	14/11/17	
REVISADO POR	FECHA	FIRMA
Leandro Ferrado	14/11/17	
APROBADO POR	FECHA	FIRMA
	14/11/17	



**Nombre del Proyecto:** Desarrollo de una aplicación Android, asistida por visión artificial, para administrar imágenes almacenadas en un dispositivo móvil

**Período del Informe:** 02/09/17 al 09/11/17

## Alcance:

Estado del proyecto				
<b>Cronograma</b>	Etapa 1: Análisis de los algoritmos a implementar y definición de requisitos para el sistema	Actividad	Fecha realización	
			Estimada	Real
		1.1 Revisión y documentación de algoritmos candidatos para el pre-procesamiento y extracción de características en imágenes.	01/08/17	02/09/17
		1.2 Revisión y documentación de técnicas para medir similitud entre imágenes procesadas y los algoritmos candidatos para el clustering.	10/08/17	13/09/17
		1.3 Definición de tecnologías y herramientas a utilizar.	23/08/17	20/09/17
		1.4 Definición de protocolo de experimentación y corpus de datos a utilizar.	25/08/17	22/09/17
		1.5 Experimentación de los algoritmos	31/08/17	05/10/17
		1.6 Documentación de los resultados de	14/09/17	20/10/17

	experimentación			
	1.7 Definición de requerimientos del proyecto.	18/09/17	25/10/17	
	1.8 Documentación de requerimientos del proyecto	28/09/17	28/10/17	
	1.9 Definición del protocolo de validación de la aplicación terminada	04/10/17	09/11/17	
<b>Riesgos</b>				
	Riesgo	Se efectivizó	Impacto	
	R002 : No disponibilidad del responsable de la ejecución del proyecto	<b>Sí</b>	Debido a que el ejecutor comenzó a trabajar, las actividades y la entrega de informe se retrasó por 4 semanas.	
	R003 : No disponibilidad de los directores del proyecto	<b>No</b>	--	
	R004: Desperfecto en el equipamiento de trabajo	<b>No</b>	--	

Proyecto Final de Carrera Ingeniería Informática	Informe de estado del proyecto	 
---	--------------------------------	---

Notas	<ul style="list-style-type: none"> <li>• Se confeccionó una Especificación de Requerimientos del Software, en cuyo apéndice se incluye información sobre los algoritmos de procesamiento de imágenes candidatos a utilizarse, el caso de aplicación, la experimentación y validación planeada sobre el mismo, recursos para llevar adelante las pruebas del proyecto y las diferentes etapas de desarrollo e implementación.</li> <li>• El link que se adjunta a continuación guarda el corpus de base de datos completo que se utilizará durante el desarrollo del proyecto para realizar el protocolo de experimentación: <a href="https://drive.google.com/drive/folders/0B-UtfbPgXdzuUGhQbWNZQmswanM?usp=sharing">https://drive.google.com/drive/folders/0B-UtfbPgXdzuUGhQbWNZQmswanM?usp=sharing</a></li> </ul>
-------	--

---

## **Especificación de requisitos de software**

**Proyecto Final de Carrera**

**Producto: IMachineApp**

**Yackel Francisco**



## Ficha del documento

Fecha	Revisión	Autor	Verificado dep. Calidad.
12/10/2017	1.0	Yackel Francisco	

Documento validado por las partes en fecha:

Por el cliente	Por el equipo de proyecto

## Contenido

<b>1</b>	<b><u>INTRODUCCIÓN</u></b>	<b>5</b>
<b>1.1</b>	<b><u>Propósito</u></b>	<b>5</b>
<b>1.2</b>	<b><u>Alcance</u></b>	<b>5</b>
<b>1.3</b>	<b><u>Personal involucrado</u></b>	<b>5</b>
<b>1.4</b>	<b><u>Definiciones</u></b>	<b>6</b>
<b>1.5</b>	<b><u>Resumen</u></b>	<b>7</b>
<b>1.6</b>	<b><u>Referencias</u></b>	<b>7</b>
<b>2</b>	<b><u>DESCRIPCIÓN GENERAL</u></b>	<b>7</b>
<b>2.1</b>	<b><u>Perspectiva del producto</u></b>	<b>7</b>
<b>2.2</b>	<b><u>Funcionalidad del producto</u></b>	<b>8</b>
<b>2.3</b>	<b><u>Características de los usuarios</u></b>	<b>8</b>
<b>2.4</b>	<b><u>Restricciones de implementación</u></b>	<b>8</b>
<b>2.5</b>	<b><u>Suposiciones y dependencias</u></b>	<b>9</b>
<b>3</b>	<b><u>REQUISITOS ESPECÍFICOS</u></b>	<b>9</b>
<b>3.1</b>	<b><u>Diagrama de casos de uso aplicación</u></b>	<b>9</b>
<b>3.2</b>	<b><u>Definición de casos de uso aplicación</u></b>	<b>10</b>
<b>3.3</b>	<b><u>Requisitos funcionales del motor de procesamiento</u></b>	<b>15</b>
3.3.1	<u>Camino de proceso de imágenes</u>	<b>15</b>
3.3.2	<u>Descripción de los pasos que conforman el motor de procesamiento</u>	<b>15</b>
<b>3.4</b>	<b><u>Requisitos no funcionales</u></b>	<b>16</b>
3.4.1	<u>Fiabilidad</u>	<b>16</b>
3.4.2	<u>Requisitos de rendimiento</u>	<b>16</b>
3.4.3	<u>Seguridad</u>	<b>16</b>

---

3.4.4	<u>Mantenibilidad</u>	17
3.4.5	<u>Portabilidad</u>	17
3.4.6	<u>Usabilidad</u>	17
<b>3.5</b>	<b><u>Historias de Usuario</u></b>	<b>17</b>
<b>4</b>	<b><u>APÉNDICES</u></b>	<b>19</b>
<b>4.1</b>	<b><u>Especificaciones de estudio y algoritmos candidatos a utilizar para el motor de procesamiento de imágenes y agrupación</u></b>	<b>19</b>
<b>4.2</b>	<b><u>Conjuntos de datos para experimentación</u></b>	<b>22</b>
<b>4.3</b>	<b><u>Protocolo de experimentación</u></b>	<b>22</b>
<b>4.4</b>	<b><u>Protocolo de validación</u></b>	<b>25</b>
<b>4.5</b>	<b><u>Recursos para realizar las pruebas del proyecto</u></b>	<b>27</b>
4.5.1	<u>Pruebas locales</u>	27
4.5.2	<u>Pruebas de la aplicación y el motor de procesamiento</u>	27
<b>4.6</b>	<b><u>Definición de etapas de desarrollo e implementación</u></b>	<b>27</b>



## Introducción

Este documento es una Especificación de Requisitos Software (ERS) realizada para otorgar al lector una comprensión total de los aspectos involucrados en el desarrollo de IMachineApp, una aplicación Android que, a partir de imágenes almacenadas en el dispositivo móvil, sugiera una estructura de directorios para organizar el conjunto, según una medida de similitud calculada entre los elementos. El mismo ha sido elaborado siguiendo el estándar IEEE Std. 830-1998.

### 1.1 Propósito

El presente documento tiene como propósito informar las decisiones de planificación y diseño del software a implementar, como así también definir las diferentes especificaciones funcionales y no funcionales. El mismo va dirigido al ejecutor del proyecto, a sus directores y a los integrantes de la cátedra de Proyecto Final de Carrera quienes actúan como evaluadores.

### 1.2 Alcance

El producto que se va a desarrollar mediante el presente proyecto se identifica como "IMachineApp", haciendo referencia en su nombre a la "máquina" que conformarán el motor de procesamiento de imágenes que usará la aplicación de fondo y el sistema de interacción con el usuario, combinado a su vez con la letra I de *image* y la palabra *app* ya que se realizará para que sea utilizada por dispositivos móviles. Aportará a la comunidad un aplicativo que sugerirá la clasificación de imágenes que se encuentran en un dispositivo según su grado de similitud o contexto, a fin de ahorrarle al usuario la realización de esta tarea de forma manual.

### 1.3 Personal involucrado

<b>Nombre</b>	Francisco Yackel
<b>Rol</b>	Analista, diseñador, desarrollador y tester
<b>Categoría Profesional</b>	Analista / Programador Junior
<b>Responsabilidad</b>	Planificación y Ejecución del Proyecto
<b>Información de contacto</b>	fyackel@gmail.com

<b>Nombre</b>	Leandro J. Ferrado
<b>Rol</b>	Co-Líder de Proyecto
<b>Categoría Profesional</b>	Jefe de Operación
<b>Responsabilidad</b>	Coordinación de Análisis y Programación
<b>Información de contacto</b>	lferrado@gmail.com

<b>Nombre</b>	Leandro Vignolo
<b>Rol</b>	Líder de Proyecto
<b>Categoría Profesional</b>	Jefe Superior
<b>Responsabilidad</b>	Dirección General del Proyecto
<b>Información de contacto</b>	ldvignolo@sinc.unl.edu.ar

## 1.4 Definiciones

- Aplicación móvil: aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles, que permite al usuario efectuar una tarea concreta de cualquier tipo —profesional, de ocio, educativas, de acceso a servicios, etc—, facilitando las gestiones o actividades a desarrollar. La misma se instala y ejecuta en sistemas operativos diseñados para un dispositivo móvil (e.g Android, iOS).
- Visión Computacional: Rama perteneciente a la inteligencia artificial la cual intenta emular en la computadora, la capacidad que tiene el cerebro de interpretar aquella información que ingresa por los ojos humanos.
- Procesamiento Digital de Imágenes: conjunto de técnicas que se aplican a las imágenes digitales con el objetivo de mejorar la calidad o facilitar la búsqueda de información.
- Reconocimiento de objetos: Conjunto de técnicas computacionales para identificar un objeto específico en una imagen digital o video. Los algoritmos de reconocimiento se basan en la comparación, el aprendizaje o los algoritmos de reconocimiento de patrones mediante técnicas basadas en la apariencia o las características.
- Reconocimiento de patrones: ciencia que se ocupa de los procesos sobre ingeniería, computación y matemáticas relacionados con objetos físicos o abstractos, con el propósito de extraer información que permita establecer propiedades entre conjuntos de dichos objetos.
- Sistema clasificador: Modelo que se construye de forma tal que ante una entrada de datos con una dimensión fija, determine a cuál de las clases disponibles pertenece. En este trabajo, dicho modelo comprende una red neuronal artificial (RNA).
- Red neuronal: Modelo computacional cuya arquitectura está basada en un conjunto de unidades o neuronas artificiales interconectadas que procesan una serie de entradas para obtener una señal de salida. Estos sistemas aprenden y se forman a sí mismos, en lugar de ser programados de forma explícita, y sobresalen en áreas donde la detección de soluciones o características es difícil de expresar con la programación convencional.
- Función clasificadora: Ubicada en la salida de la red, establece una predicción de clasificación sobre un dato de entrada.
- Patrón / Ejemplo: Dato singular que tiene igual dimensión que la de la entrada del modelo. Sirven para “alimentar” al modelo, ya sea para entrenarlo o validarlo.
- Aprendizaje supervisado: Paradigma por el cual el modelo se entrena mediante

ejemplos con etiquetas, y se trata de minimizar un error entre la salida calculada y la deseada.

- Aprendizaje no supervisado: Paradigma por el cual el modelo no utiliza etiquetas para los patrones de entrenamiento, con lo cual se pueden extraer rasgos o características de los mismos.
- Algoritmo de agrupamiento o “clustering”: Sistema de aprendizaje no supervisado cuya tarea se basa en agrupar ejemplos de un conjunto de entrada basándose en un concepto o función de “similitud” computada sobre las características de cada ejemplo.
- Función de similitud: modelo matemático compuesto por las diferentes características extraídas de la imagen actuando como variables independientes, encargado de arrojar un resultado que sea útil para poder realizar comparaciones.

## 1.5 Resumen

Este documento consta de cuatro secciones. En la primera sección se realiza una introducción al mismo y se proporciona una visión general acerca de lo que trata el producto y definiciones necesarias para su entendimiento. En la segunda sección se realiza una descripción general del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados, las tecnologías a utilizar. La tercera sección del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema, puntualizando la prioridad asignada a cada uno. Por último, la cuarta sección es de apéndices, donde se detallan algoritmos candidatos a utilizar para el motor de procesamiento, los casos de aplicación elegidos y cómo se experimentará sobre ellos utilizando el producto de software desarrollado.

## 1.6 Referencias

- ❖ Méndez, G. (2008). Especificación de Requisitos según el estándar de IEEE 830. Facultad de Informática, Universidad Complutense de Madrid.
- ❖ Rafael C. González and Richard E. Woods, Digital Image Processing, Prentice-Hall, 3rd. ed. (2008)
- ❖ Baggio, D. L. (2012). Mastering OpenCV with practical computer vision projects. Packt Publishing Ltd.
- ❖ Liu, J., & Yu, J. (2011, November). Research on development of android applications. In Intelligent Networks and Intelligent Systems (ICINIS), 2011 4th International Conference on (pp. 69-72). IEEE.

# 2 Descripción general

## 2.1 Perspectiva del producto

El producto es de carácter independiente, y pensado para servir como aplicación que agrupe, de manera automática, las imágenes de un directorio seleccionado por el usuario con el fin de sugerirle una organización de dichas imágenes basada en las similitudes que

se computaron sobre ellas. Luego, una vez realizado el procesamiento, éste podrá tomar la decisión que desee interactuando con el resultado (e.g eliminar un grupo obtenido, fusionarlo con otros, renombrar carpetas, ingresar en un grupo y eliminar alguna imagen en particular, etc.). Para ello el sistema consta de dos componentes principales: el motor que realiza el agrupamiento de las imágenes, y la aplicación móvil que utiliza el motor para gestionar la organización de las imágenes en el dispositivo. Se define un único perfil de acceso al producto, el de usuario de la aplicación quien con conocimientos básicos de manejo de aplicaciones móviles puede interactuar con el producto sin necesidad de algún conocimiento o capacitación adicional.

## 2.2 Funcionalidad del producto

La aplicación en funcionamiento dispondrá de las siguientes características:

- Agrupar automáticamente imágenes según su grado de similitud mediante visión computacional.
- Permitir elegir directorios específicos a procesar.
- Acceder a las imágenes del dispositivo para procesarlas.
- No necesita conexión a internet.
- Crear y modificar directorios en el dispositivo.
- Permitir combinar y eliminar directorios obtenidos por el proceso.

## 2.3 Características de los usuarios

<b>Tipo de usuario</b>	Usuario final
<b>Formación</b>	Ninguna
<b>Habilidades</b>	Conocimientos en general sobre el uso de aplicaciones que corren bajo plataforma Android
<b>Actividades</b>	Uso de la aplicación

## 2.4 Restricciones de implementación

- Aplicación desarrollada para el sistema operativo Android
- Lenguajes y tecnologías en uso: XML 1.1, JAVA 1.8.0\_131, C++ 11, Python 2.7.
- La aplicación funcionará para versiones de Android KitKat (4.4) en adelante.
- Los paradigmas de programación a utilizar serán: Orientado a objetos, Imperativo
- El control de versionado de código será gestionado por Git 2.7.4
- El sistema deberá intuitivo y fácil de utilizar, a fin de que pueda aprovecharse por la mayor cantidad de personas.

## 2.5 Suposiciones y dependencias

Se supone que todo lo detallado en esta versión de la ERS ya es suficientemente claro y estable como para desarrollar el producto sin inconvenientes. El resto de modificaciones que pueda sufrir este documento serán de carácter aditivo y/o aclarativo, y no alterarán de manera destructiva el desarrollo en ejecución. En cuanto a las dependencias de tecnologías definidas, se establece un detalle de la versión utilizada en cada una para evitar posibles incompatibilidades.

Como evolución o trabajo a futuro se plantea la posibilidad de mejorar la experiencia de usuario respecto a las interfaces, en base al *feedback* colectado de aquellas personas que la utilicen una vez que se despliegue. Otra alternativa es que se vaya personalizando con el uso a través de diferentes consultas que mantendría la aplicación con el usuario, y así lograr mejores aproximaciones y resultados al paso del tiempo. Además, existe la posibilidad de extender la app a otros sistemas operativos (iOS, Window), y también hacer una versión escritorio y una web para aquellas imágenes almacenadas directamente en la computadora. Por último, existiría la posibilidad de integrar el resultado con algún producto de almacenamiento en la nube salvaguardando el resultado fácilmente.

## 3 Requisitos específicos

### 3.1 Diagrama de casos de uso aplicación

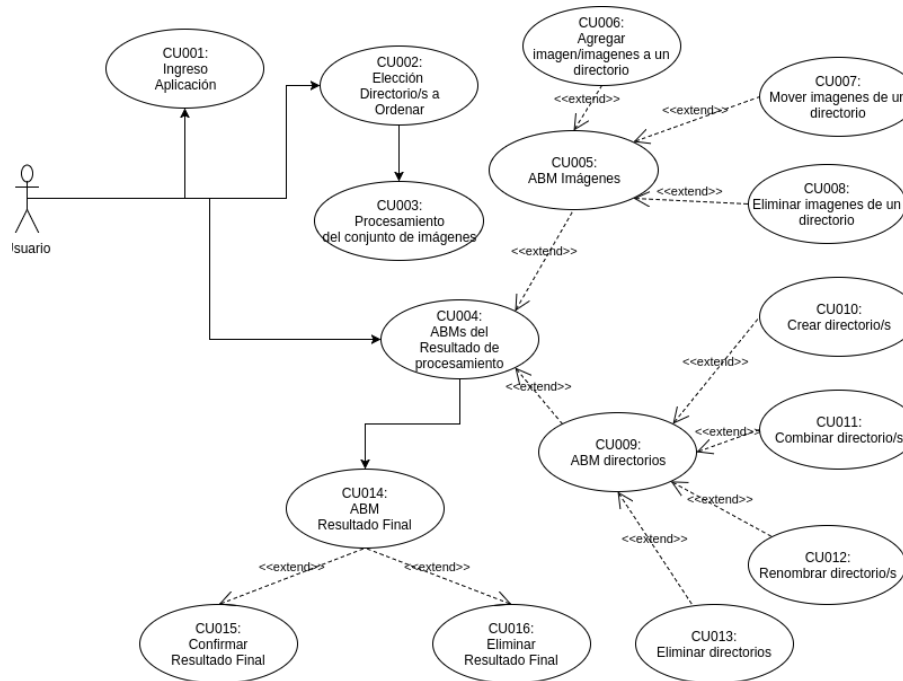


Fig 1. Diagrama de Casos de Uso

### 3.2 Definición de casos de uso aplicación

<b>Caso de uso</b>	<b>CU001: Ingreso al sistema</b>
<b>Actores</b>	Usuario Final
<b>Tipo</b>	Primario
<b>Referencias</b>	-
<b>Precondición</b>	Tener la aplicación descargada en el dispositivo
<b>Postcondición</b>	Presentación del menú de funcionalidades que permite utilizar la aplicación
<b>Descripción</b>	El usuario podrá elegir cualquiera de las opciones que presenta el inicio de la aplicación: realizar un agrupamiento de imágenes, elegir un directorio en particular, etc.
<b>Resumen</b>	Primer interfaz gráfica que ve el usuario al iniciar la aplicación, donde elige qué funcionalidad utilizar.

<b>Caso de uso</b>	<b>CU002: Elección directorio/s a ordenar</b>
<b>Actores</b>	Usuario Final
<b>Tipo</b>	Opcional
<b>Referencias</b>	CU001
<b>Precondición</b>	Haber seleccionado la opción de ordenar un nuevo conjunto de imágenes.
<b>Postcondición</b>	Dar la opción al usuario de elegir un directorio específico sobre el cual realizar el ordenamiento de imágenes o bien ordenar todos los directorios contenedores de imágenes juntos.
<b>Descripción</b>	El usuario puede decidir elegir un directorio en particular (i.e: Imágenes de WhatsApp) o bien un grupo de directorios (i.e: Imágenes de WhatsApp y Fotos de la Cámara), o que el sistema busque todas las imágenes que se encuentran en el dispositivo.
<b>Resumen</b>	Opción disponible a elegir por el usuario donde decidirá qué directorio contenedor de imágenes mandará a procesar para que las agrupe de manera inteligente.

<b>Caso de uso</b>	<b>CU003: Procesamiento del conjunto de imágenes</b>
<b>Actores</b>	-
<b>Tipo</b>	Primario
<b>Referencias</b>	CU001 CU002
<b>Precondición</b>	Haber seleccionado un conjunto de directorios en particular o bien la opción de procesar todas las imágenes del

	dispositivo.
<b>Postcondición</b>	Informar el estado del procesamiento al usuario a medida que se va realizando
<b>Descripción</b>	Ejecución del motor de procesamiento y agrupación de imágenes según su grado de similitud o contexto
<b>Resumen</b>	El sistema dará uso al motor de procesamiento a fin de que sus imágenes puedan agruparse de forma inteligente. A su vez, informar el estado de ejecución conforme avanza el procesamiento.

<b>Caso de uso</b>	<b>CU004: ABMs Resultado de Procesamiento</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Primario
<b>Referencias</b>	CU003
<b>Precondición</b>	El CU003 tuvo que haber terminado exitosamente para este lote o grupo de directorios que contienen las imágenes procesadas.
<b>Postcondición</b>	Mostrar las opciones de administración disponibles a utilizar para el grupo de directorios e imágenes.
<b>Descripción</b>	Se podrán realizar diferentes tareas de administración sobre el lote de directorios que contienen las imágenes procesadas a fin de que el resultado final quede como el usuario final desea.
<b>Resumen</b>	El usuario podrá realizar tareas de administración sobre el la salida del conjunto procesado a fin de poder lograr el resultado final esperado.

<b>Caso de uso</b>	<b>CU005: ABM Imágenes</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Opcional
<b>Referencias</b>	CU004
<b>Precondición</b>	El CU003 tuvo que haber terminado exitosamente para poder realizar tarea administración sobre imágenes
<b>Postcondición</b>	Mostrar las opciones de administración disponibles a utilizar para el grupo de imágenes procesadas.
<b>Descripción</b>	Se podrán realizar diferentes tareas de administración sobre las imágenes a fin de que el resultado final quede como el usuario desea.
<b>Resumen</b>	El usuario podrá realizar tareas de administración sobre el conjunto de imágenes procesadas a fin de poder lograr el resultado final esperado.

<b>Caso de uso</b>	<b>CU006: Agregar imagen/imágenes a un directorio</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU005
<b>Precondición</b>	El usuario desee agregar una imagen.
<b>Postcondición</b>	Mostrar la opción de agregar una imagen al directorio seleccionado.
<b>Descripción</b>	Opción que permite al usuario agregar una imagen o un conjunto de imágenes a un directorio.
<b>Resumen</b>	El usuario podrá agregar imágenes que no estén en un directorio a través de esta opción de administración.

<b>Caso de uso</b>	<b>CU007: Mover imágenes de un directorio</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU005
<b>Precondición</b>	El usuario desee mover una imagen.
<b>Postcondición</b>	Mostrar la opción de mover una imagen seleccionada.
<b>Descripción</b>	Opción que permite al usuario mover una imagen o un conjunto de imágenes a un directorio determinado.
<b>Resumen</b>	El usuario podrá mover imágenes de un directorio a otro a través de esta opción de administración.

<b>Caso de uso</b>	<b>CU008: Eliminar imágenes de un directorio</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU005
<b>Precondición</b>	El usuario desee eliminar una imagen.
<b>Postcondición</b>	Mostrar la opción de eliminar una imagen o un conjunto de imágenes seleccionadas.
<b>Descripción</b>	Opción que permite al usuario eliminar una imagen o un conjunto de imágenes de un directorio determinado.
<b>Resumen</b>	El usuario podrá eliminar imágenes de un directorio a través de esta opción de administración.

<b>Caso de uso</b>	<b>CU009: ABM Directorios</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Opcional
<b>Referencias</b>	CU004
<b>Precondición</b>	El CU003 tuvo que haber terminado exitosamente para poder realizar tarea administración sobre imágenes



<b>Postcondición</b>	Mostrar las opciones de administración disponibles a utilizar para el grupo de directorios generados.
<b>Descripción</b>	Se podrán realizar diferentes tareas de administración sobre los directorios a fin de que el resultado final quede como el usuario desea.
<b>Resumen</b>	El usuario podrá realizar tareas de administración sobre el conjunto de directorios generados a fin de poder lograr el resultado final esperado.

<b>Caso de uso</b>	<b>CU010: Crear directorio/s</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU004
<b>Precondición</b>	El usuario desee crear un directorio nuevo.
<b>Postcondición</b>	Mostrar la opción de administración referida a la creación de directorios.
<b>Descripción</b>	Opción que permite al usuario crear un directorio nuevo el cual se encontrará por defecto vacío.
<b>Resumen</b>	El usuario podrá crear directorios a fin de realizar una organización subjetiva.

<b>Caso de uso</b>	<b>CU011: Combinar directorio/s</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU004
<b>Precondición</b>	El usuario desee combinar un conjunto de directorios.
<b>Postcondición</b>	Mostrar la opción de administración referida a la combinación de directorios.
<b>Descripción</b>	Opción que permite al usuario crear un directorio nuevo el cual se encontrará compuesto por dos o más directorios previos.
<b>Resumen</b>	El usuario podrá combinar directorios a fin de realizar una organización subjetiva.

<b>Caso de uso</b>	<b>CU012: Renombrar directorio/s</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU004
<b>Precondición</b>	El usuario desee renombrar un directorio.
<b>Postcondición</b>	Mostrar la opción de administración referida al cambio de nombre de un directorio.
<b>Descripción</b>	Opción que permite al usuario cambiar el nombre de un

	directorio.
<b>Resumen</b>	El usuario podrá cambiar el nombre de un/unos directorio/s.

<b>Caso de uso</b>	<b>CU013: Eliminar directorio/s</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU004
<b>Precondición</b>	El usuario desee eliminar un/unos directorio/s.
<b>Postcondición</b>	Mostrar la opción de administración referida a la eliminación de un directorio o conjunto de directorios.
<b>Descripción</b>	Opción que permite al usuario eliminar uno o más directorios.
<b>Resumen</b>	El usuario podrá eliminar un directorio o un conjunto de directorios junto a todas las imágenes procesadas que estén contenidas.

<b>Caso de uso</b>	<b>CU014: ABM Resultado Final</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Primario
<b>Referencias</b>	CU004
<b>Precondición</b>	Debe existir al menos un resultado de procesamiento registrado
<b>Postcondición</b>	Mostrar las opciones de administración disponibles a utilizar para el resultado final logrado.
<b>Descripción</b>	Se podrán realizar diferentes tareas de administración sobre el resultado final de imágenes procesadas y administradas por el usuario en la etapa anterior a fin de concluir con la tarea.
<b>Resumen</b>	El usuario podrá realizar tareas de administración sobre el resultado final a fin de poder concluir con el uso de la aplicación para ese momento determinado.

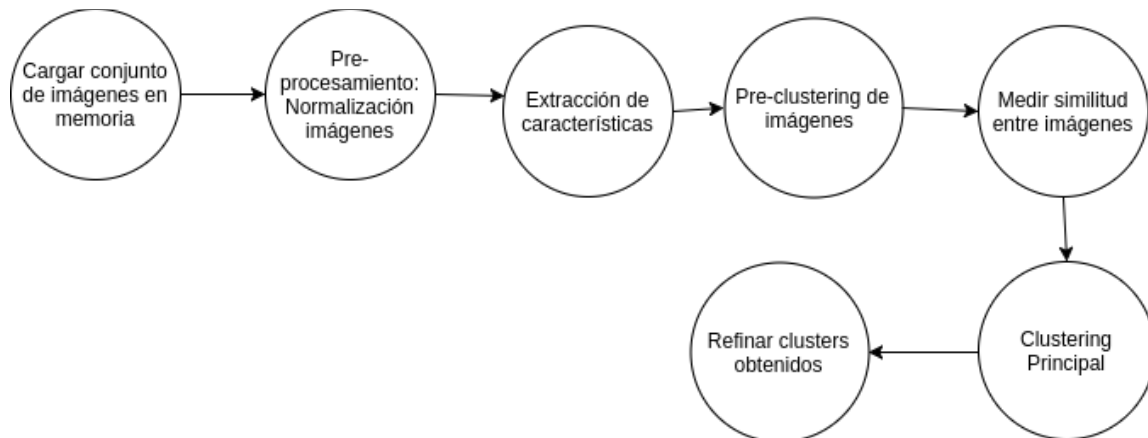
<b>Caso de uso</b>	<b>CU0015: Confirmar Resultado Final</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU004
<b>Precondición</b>	Debe existir al menos un resultado de procesamiento registrado
<b>Postcondición</b>	Generar los diferentes directorios en el dispositivo
<b>Descripción</b>	Se podrán confirmar los resultados a fin de guardar los cambios.
<b>Resumen</b>	El usuario podrá confirmar el resultado final para que los directorios se creen en el teléfono y así poder guardar todos

	los cambios realizados.
--	-------------------------

<b>Caso de uso</b>	<b>CU0016: Eliminar Resultado Final</b>
<b>Actores</b>	Usuario final
<b>Tipo</b>	Secundario
<b>Referencias</b>	CU004
<b>Precondición</b>	Debe existir al menos un resultado de procesamiento registrado
<b>Postcondición</b>	Preguntar si realmente desea perder todos los cambios
<b>Descripción</b>	Se podrá eliminar el resultado final a fin de no guardar los cambios y procesamiento realizado.
<b>Resumen</b>	El usuario podrá eliminar el resultado final para que los directorios no se creen en el teléfono y así seguir teniendo sus imágenes tal cual se encontraban antes de interactuar con la aplicación.

### 3.3 Requisitos funcionales del motor de procesamiento

#### 3.3.1 Camino de proceso de imágenes



#### 3.3.2 Descripción de los pasos que conforman el motor de procesamiento

1. Carga conjunto de imágenes en memoria: Las imágenes deberán ser cargadas en memoria a fin de poder realizar en las mismas los diferentes procesos necesarios.
2. Normalización de imágenes: utilización de diferentes técnicas pertenecientes al procesamiento digital de imágenes a fin de normalizar el

conjunto de imágenes eliminando ruido e interferencia de las mismas para mejorar el aspecto y la calidad.

3. Extracción de características: utilización de técnicas pertenecientes a la visión computacional y al aprendizaje maquina para poder extraer aquellas características que hacen distintiva cada imagen, con el fin de obtener una representación vectorial o matricial adecuada de dimensión fija.
4. Pre-clustering de imágenes (opcional): Detección y separación del estudio de aquellas imágenes que parezcan no pertenecer a una secuencia fotográfica y por ende no tenga sentido clusterizar, a fin de utilizar menos recursos computacionales durante el clusterizado principal. Se evaluará su desempeño a fin de decidir si vale la pena agregarlo al motor de procesamiento.
5. Medir similitud entre imágenes: Comparar las representaciones obtenidas por cada imagen, en términos de una función analítica adecuada, para medir la afinidad a que estén o no en un mismo grupo luego del clustering.
6. Clustering principal: utilizar diferentes técnicas del aprendizaje maquina a fin de encontrar los diferentes grupos de imágenes que más se correlacionan entre sí.
7. Refinar clusters obtenidos (opcional): evaluar la posibilidad de realizar un postproceso a los diferentes grupos obtenidos en la etapa anterior a fin de encontrar una solución más acertada.

## 3.4 Requisitos no funcionales

### 3.4.1 Fiabilidad

- El sistema debe tener una interfaz de uso intuitiva y sencilla.
- La interfaz de usuario debe ajustarse a las características generales con la que cuentan la mayoría de los dispositivos móviles actuales.

### 3.4.2 Requisitos de rendimiento

- Garantizar que el procesamiento de información se mantenga dentro de un tiempo no mayor a los 2 minutos, detallando el estado de proceso constantemente.
- La capacidad máxima de imágenes a procesar en un período será de 1000 imágenes.

### 3.4.3 Seguridad

- Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información siempre se mantendrá en el teléfono sin la necesidad de realizar transferencias a través de internet.
- Garantizar la seguridad del sistema con respecto a la información y datos que se manejan, asegurándose que ante un error de proceso el usuario no pierda sus archivos.

### 3.4.4 Mantenibilidad

- El sistema debe disponer de documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible.

### 3.4.5 Portabilidad

- El sistema será implantado bajo la plataforma de Android.

### 3.4.6 Usabilidad

- El sistema debe contar con un manual de usuario estructurado adecuadamente.
- El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.
- El sistema debe poseer interfaces gráficas bien formadas.

## 3.5 Historias de Usuario

Historia de Usuario 1	
Usuario	Usuario final
Nombre historia:	Ingresar al sistema
Prioridad en negocio:	Alta
Riesgo en desarrollo:	Baja
Puntos estimados:	2
Descripción	Yo como usuario final quiero ingresar al sistema para realizar una acción.
Validación	El usuario final dispondrá de un icono representativo de la

	aplicación para poder ingresar al sistema
--	---

Historia de Usuario 2	
<b>Usuario</b>	Usuario final
<b>Nombre historia:</b>	Elegir un directorio
<b>Prioridad en negocio:</b>	Alta
<b>Riesgo en desarrollo:</b>	Baja
<b>Puntos estimados:</b>	2
<b>Descripción</b>	Yo como usuario final quiero elegir el o los directorios donde almaceno imágenes para mandar a procesar por la aplicación.
<b>Validación</b>	El usuario final dispondrá de una interface donde podrá seleccionar las diferentes carpetas contenedoras de imágenes o bien un check a fin de que la aplicación busque todas las imágenes almacenadas en el dispositivo móvil.

Historia de Usuario 3	
<b>Usuario</b>	Usuario final
<b>Nombre historia:</b>	Saber el estado del proceso
<b>Prioridad en negocio:</b>	Alta
<b>Riesgo en desarrollo:</b>	Medio
<b>Puntos estimados:</b>	5
<b>Descripción</b>	Yo como usuario final quiero saber en qué porcentaje de procesamiento se encuentra la aplicación para mantenerme informado.
<b>Validación</b>	Durante el procesamiento de imágenes se mantendrá informado al usuario a través de un gráfico para que pueda saber cuánto resta por procesar.

Historia de Usuario 4	
<b>Usuario</b>	Usuario final
<b>Nombre historia:</b>	Administrar imágenes de un directorio
<b>Prioridad en negocio:</b>	Alta
<b>Riesgo en desarrollo:</b>	Alta
<b>Puntos estimados:</b>	12
<b>Descripción</b>	Yo como usuario final quiero agregar, mover o eliminar fotos de un directorio para organizar mejor mis imágenes.
<b>Validación</b>	El usuario final dispondrá de una sección donde podrá interactuar con el resultado de procesamiento a fin de darle una organización final a todas las carpetas sugeridas por la aplicación.

Historia de Usuario 5
-----------------------

<b>Usuario</b>	Usuario final
<b>Nombre historia:</b>	Administrar resultado final
<b>Prioridad en negocio:</b>	Alta
<b>Riesgo en desarrollo:</b>	Alta
<b>Puntos estimados:</b>	12
<b>Descripción</b>	Yo como usuario final quiero guardar o eliminar el resultado final arrojado por la aplicación para obtener una mejor organización de mis imágenes.
<b>Validación</b>	El sistema proveerá al usuario de una pantalla donde podrá realizar tanto la operación de guardado como de descarte del resultado final arrojado por la aplicación.

## 4 Apéndices

### 4.1 Especificaciones de estudio y algoritmos candidatos a utilizar para el motor de procesamiento de imágenes y agrupación

Con el objetivo de que la aplicación pueda tener un buen desempeño y realizar satisfactoriamente la tarea para la cual fue pensada, se definen un conjunto de especificaciones de estudio y algoritmos candidatos a ser tenidos en cuenta en cada una de las etapas por las que pasarán las imágenes, desde que se las carga en memoria hasta la determinación del grupo al que pertenecerán cada una:

1. Cargar el conjunto de imágenes en memoria:
  - Espacio de colores: RGB - HSV - Escala Grises
2. Pre-procesamiento de imágenes:
  - Redimensión de imágenes (resize): a fin de que todas tengan el mismo tamaño.
  - Segmentación: a fin de simplificar la representación de la imagen.
  - Normalización de píxeles: a fin de poder realizar comparaciones.
  - Filtrado: a fin de eliminar impurezas o basura en las imágenes
  - Ecualización: a fin de obtener el histograma de cada imagen con una distribución uniforme para mejorar el contraste de la misma.
3. Extracción de características de cada imagen:
  - Algoritmos basados en descriptores:
    - BRISK - BRIEF - ORB [1]: Realizan detección de características locales de la imagen con el objetivo de realizar comparaciones.
    - Restricción: No usar algoritmos patentados como SIFT o SURF
  - Algoritmos basados en texturas:
    - LBP - HAAR - HOG [2]: Métodos utilizados para clasificación de imágenes a partir de histogramas, basados en diferencias computadas entre píxeles comprendidos en secciones extraídas de la imagen.
  - Modelos pre-entrenados para extracción de características en imágenes:
    - Red neuronal con parámetros ajustados sobre un gran conjunto de imágenes [3], de la cual se puede utilizar tanto su predicción en términos de clasificación como la

activación de la capa previa a la clasificación (usualmente, utilizada como vector de características para modelos supervisados).

4. Pre-clustering (opcional):

- Separar del estudio aquellas imágenes que parezcan no pertenecer a una secuencia fotográfica y por ende no tenga sentido clusterizar.
  - Podrían ser memes, screenshots, etc.
  - Evaluar la posibilidad de que el usuario lo solicite mediante una opción del sistema
  - Podría validarse por el usuario antes de continuar con el clustering principal.
- Clasificador:
  - Utilizar clases de un clasificador entrenado con imágenes, donde las clases seleccionadas se corresponden a categorías genéricas que puedan interpretarse como clusters estándar (como el caso de imágenes de memes y de screenshots).
  - Otras heurísticas posibles (opcional):
    1. Detectar texto en imágenes y verificar que porción de la imagen cubre para distinguir imágenes artificiales de aquellas naturales.
    2. Imágenes con gran frecuencia de pocos colores: fotos reales tienen mayor gama de colores que memes o screenshots.
  - Evaluar si conviene computar meta clusters a fin de poder agrupar menos puntos:
    1. Local Sensitive Hashing (LSH) [4]: algoritmo utilizado para reducir dimensionalidad de los vectores de características y favorecer el agrupamiento de aquellos que sean similares en dicho espacio reducido.
    2. Canopy clustering [5]: algoritmo no supervisado de pre-clustering que tiene como objetivo acelerar las operaciones de agrupamiento de grandes conjuntos de datos, donde el uso directo de otro algoritmo puede resultar poco práctico debido al tamaño del conjunto de datos.

5. Computar similitud entre imágenes basadas en características extraídas:

- Basados en distancia entre histogramas, normalizando las variables cuantizadas en los histogramas a fin de que puedan ser comparables.
  - Funciones de ejemplo: Bhattacharyya [6], intersección de histogramas, etc.
- Correlación o distancia entre vectores.
- FLANN - Annoy [7]: Métodos para comparar y reunir descriptores similares.

6. Clustering principal - algoritmos candidatos y medidas de similitud:

- Fuzzy clustering [9]: clase de algoritmos de agrupamiento donde cada elemento tiene un grado de pertenencia difuso a los grupos.
  - Membresía: grados de membresía son asignados a cada punto de datos. Estos grados indican en qué porcentaje cada punto pertenece a un grupo. Por lo tanto, puntos en el borde del cluster, se encontrarán con un grado menor que los puntos en el centro del cluster, debido a su distancia al centro.



- Fuzzy c-means [10]
- Graph Clustering [11]: basados en encontrar conjuntos de vértices "relacionados" en los gráficos.
  - Markov Cluster (MCL) [12]: Algoritmo no supervisado de clusterizado rápido y escalable utilizado para grafos, basado en la simulación del flujo estocástico en grafos.
  - Affinity propagation [8]: Algoritmo de agrupamiento el cual no requiere que el número de clusters sea determinado o estimado de forma previa.

7. Evaluar posibilidad de refinar clusters obtenidos.

- Por ejemplo, agrupando aquellos grupos cuyo valor de Silhouette score indique un alto grado de cohesión.
- Se puede procesar nuevamente algún grupo que se considere demasiado grande.
- Aquellos grupos con muy pocos ejemplos se pueden reunir en un único grupo, informando que no se pudieron asociar con el resto de las imágenes.

Cabe aclarar que las diferentes opciones presentadas anteriormente no son de carácter definitivo y podría utilizarse algún algoritmo no detallado con resultados mejores que surja en la etapa de desarrollo.

[1] Miksik, O., & Mikolajczyk, K. (2012, November). Evaluation of local detectors and descriptors for fast feature matching. In Pattern Recognition (ICPR), 2012 21st International Conference on (pp. 2681-2684). IEEE.

[2] Kadir, K., Kamaruddin, M. K., Nasir, H., Safie, S. I., & Bakti, Z. A. K. (2014, August). A comparative study between LBP and Haar-like features for Face Detection using OpenCV. In Engineering Technology and Technopreneuship (ICE2T), 2014 4th International Conference on (pp. 335-339). IEEE.

[3] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[4] Datar, M., Immorlica, N., Indyk, P., & Mirrokni, V. S. (2004, June). Locality-sensitive hashing scheme based on p-stable distributions. In Proceedings of the twentieth annual symposium on Computational geometry (pp. 253-262). ACM.

[5] McCallum, A., Nigam, K., & Ungar, L. H. (2000, August). Efficient clustering of high-dimensional data sets with application to reference matching. In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 169-178). ACM.

[6] Choi, E., & Lee, C. (2003). Feature extraction based on the Bhattacharyya distance. Pattern Recognition, 36(8), 1703-1709.

[7] Geoghegan, C. (2015). Improving learning resource recommendations for students.

[8] Frey, B. J., & Dueck, D. (2007). Clustering by passing messages between data points. science, 315(5814), 972-976.

- [9] Xie, X. L., & Beni, G. (1991). A validity measure for fuzzy clustering. IEEE Transactions on pattern analysis and machine intelligence, 13(8), 841-847.
- [10] Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. Computers & Geosciences, 10(2-3), 191-203.
- [11] Schaeffer, S. E. (2007). Graph clustering. Computer science review, 1(1), 27-64.
- [12] Van Dongen, S. M. (2001). Graph clustering by flow simulation (Doctoral dissertation).

## 4.2 Conjuntos de datos para experimentación

A fines de validar la utilidad y potencialidad del software, se define un conjunto de base de datos de prueba compuesto por diversas imágenes que provienen de lugares diferentes y bajo normativas distintas. La primera de ellas consiste en la carpeta de imágenes de WhatsApp del ejecutor del proyecto, que contando con un total de 1700 imágenes aproximadamente, se considera representativa para cualquier usuario promedio. La segunda, 2000 imágenes pertenecientes al portal de internet 9gag[1], sitio donde la comunidad participe sube imágenes de todo tipo de temática, todos los días. Por último, se utilizó también una base de datos denominada VOC[2], la cual cuenta con diversos grupos de imágenes estandarizadas y se utiliza para medir algoritmos de clasificación de imágenes en diferentes competencias que se realizan año a año, sumando más de 17000 imágenes.

Sumando todos los directorios, se contabilizan más de 20.000 imágenes pertenecientes a la base de datos de prueba.

[1] <https://9gag.com/>

[2] <http://host.robots.ox.ac.uk/pascal/VOC/>

## 4.3 Protocolo de experimentación

El objetivo de esta sección es especificar las pruebas que deberán ser ejecutadas sobre el motor de clustering, con el fin de validar que la calidad del agrupamiento de imágenes obtenida en su uso será aceptable como para ser de utilidad en la tarea de organización de las mismas.

La experimentación deberá abarcar los siguientes aspectos:

- **Datos:** Se utilizarán las bases de datos señaladas anteriormente, de la siguiente forma:
  - Se hará un muestreo de grupos o “clusters” a utilizar en la prueba de cada base de datos, basándose en las clases o etiquetas asociadas a cada imagen. Dicho procedimiento se especifica más adelante.
  - El total de imágenes a utilizar para cada experimento debe variar entre 10 y 250 imágenes para las pruebas normales, y entre 250 y 1000 imágenes para las pruebas de escalabilidad.

- Muestreo de clusters: Se realiza a fin de obtener el conjunto de datos con el cual validar el motor de clustering. Para ello se realizan los siguientes pasos:
  - Se elige la cantidad de clusters a obtener (entre 2 y 50)
  - Se muestrea una selección de clases a tratar, una por cada cluster a obtener, abarcando todas las bases de datos disponibles. Es deseable que dichas clases sean en su mayoría bastante distintivas entre sí para que los resultados sean representativos.
  - Por cada una de las clases, se muestrea un total de imágenes para cada cluster, que puede variar entre 5 o 10 imágenes, hasta completar el total de imágenes a agrupar.
  - Se etiqueta cada imagen con su respectivo cluster esperable, a fin de tener un mecanismo supervisado para validar los clusters obtenidos con el motor
- Pre-procesamiento: Con el fin de normalizar las condiciones del total de imagen a clusterizar, se disponen de los siguientes pasos:
  - Escalar imágenes a una resolución fija (menor a 640x480) para que sea computacionalmente más manejable.
  - Se puede optar por aplicar algunas de las técnicas de pre-procesamiento ya mencionadas para mejorar la calidad de las imágenes a tratar.
- Extracción de características: Deben basarse tanto en el aspecto (en término de formas y/o bordes) como en los colores de cada imagen. Deben resultar en un vector unidimensional, por lo cual cada imagen se puede representar por un vector de no más de 1000 componentes para que sea manejable computacionalmente.
  - Funciones de similitud: Deben estar basadas en comparación de vectores, contemplando la naturaleza o forma en que se computaron dichos vectores (por ejemplo, si son histogramas entonces la similitud puede definirse como una función de divergencia). Es preferente que esta métrica tenga las propiedades de ser simétrica y positiva para un manejo más práctico.
  - En caso de utilizar múltiples métodos de extracción de características, se debe realizar una ponderación de los valores de similitud obtenidos en cada caso para que el algoritmo de clustering maneje un único valor representativo.
- Tarea: Agrupar imágenes basándose en un concepto de similitud computada sobre pares de las mismas. Para ello se tiene que cumplir con las siguientes especificaciones:
  - Se tiene preferencia por los algoritmos basados en “graph clustering”, de forma tal que se pueda computar similitud entre pares de imágenes y además no se deba especificar la cantidad de clusters a obtener (e.g. K-means) ya que no es el caso de uso en la App.
  - Opcional: algoritmo de pre-clustering para reducir la cantidad de imágenes a procesar en el clustering final, al agrupar en una primera instancia aquellos puntos que son muy similares (o bien repetidos) y no tiene sentido comparar contra el resto de las imágenes.
- Métricas: Dado que la forma de obtener los datos de pruebas está basado en clases, se considera la evaluación en términos de la clasificación respecto a los datos de pruebas así como también utilizando métricas de clustering.
- Clasificación [1]:

- **F1-Score:** computado como la media armónica de los valores de "precision" y "recall" en la clasificación, extendido para soportar múltiples clases mediante "macro-averaging". El valor resultante varía entre 0 (clasificación pésima) y 1 (clasificación perfecta).
  - **Accuracy:** calculada como la exactitud de la clasificación, dada por la cantidad de aciertos sobre el total. El valor resultante varía entre 0 (clasificación pésima) y 1 (clasificación perfecta).
- Clustering [2][3]:
  - **V-Measure:** computado como la media armónica de los valores de "homogeneidad" y "completitud" en el clustering. Para ello se necesita tener un "ground truth" de clusters esperados para comparar con el resultado obtenido. El valor resultante varía entre 0 (clustering pésimo) y 1 (clustering perfecto).
  - **Silhouette:** medida de cuán similares son los ítems dentro de un cluster (cohesión) comparados con los de otros clusters (separación). El valor resultante varía entre -1 (clustering pésimo) y 1 (clustering perfecto), donde un valor de 0 indica solapamiento entre clusters.
- Configuraciones: Por cada experimento, se documentará la variación de los parámetros utilizados en cada algoritmo (tanto de clustering como de extracción de características), y los resultados obtenidos en cada prueba, a fines de conocer cuál es la configuración más estable y precisa a establecer en el motor que se integrará en la aplicación.
- Cantidad de pruebas: Se sugiere realizar entre 5 y 20 ejecuciones por cada configuración establecida, variando el conjunto de imágenes muestreadas a clusterizar, y promediando las métricas para tener una idea general del desempeño obtenido.
- Desempeño computacional: Se debe medir el tiempo transcurrido en cada prueba ejecutada sobre el ordenador personal, para así tener una idea del esfuerzo computacional que requiere la metodología en términos de procesamiento (sin tener en cuenta la memoria utilizada, cuyo umbral máximo se supone fijo para las pruebas a realizar).
- Criterio de aprobación: Dado que en una aplicación real la valoración de un cluster puede ser subjetiva (es decir, el juicio depende mucho del usuario final), los niveles de aceptación en las métricas se relajan dado que el clustering final obtenido pretende ser únicamente una sugerencia de organización que el usuario puede modificar con menor esfuerzo que si lo hiciera con el conjunto total de imágenes. Por lo tanto, se establecen los siguientes criterios:
  - Métricas: Se sugieren ciertos umbrales para guiar el juicio en forma objetiva respecto a la elección de los algoritmos y sus configuraciones, aunque no es estrictamente necesario igualar o superarlos en su totalidad para las pruebas a ejecutar. Los valores de las métricas a utilizar deben ser obtenidos al promediar los resultados de distintas ejecuciones para distintos conjuntos de datos muestreados, tal como se mencionó en el ítem anterior. Por lo tanto, los umbrales sugeridos son:
    - F1-Score > 0.4
    - Accuracy > 1 / (cantidad de clases tratadas)
    - V-measure > 0.5
    - Silhouette > 0

- Valoración subjetiva del clustering obtenido: para saber si le es útil al usuario en un caso real (entre 1 y 5 estrellas). Realizado por hasta 5 usuarios distintos.
- En cuanto al desempeño en términos computacionales, se espera que el procesamiento total no lleve más de 150 segundos para las pruebas normales (hasta 250 imágenes) y no más de 300 segundos para las pruebas de escalabilidad (hasta 1000 imágenes).

Dado que no se tiene como objetivo tener un desempeño específico, en términos de la calidad de clustering, se establece que no es necesario realizar un análisis que se compare o compita con el estado del arte en dicha tarea ya que el motor no se pretende más que para sugerir grupos de imágenes que el usuario puede incluso modificar.

[1] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427-437

[2] Rosenberg, A., & Hirschberg, J. (2007, June). V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. In *EMNLP-CoNLL* (Vol. 7, pp. 410-420).

[3] Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20, 53-65.

## 4.4 Protocolo de validación

A continuación se definirán un conjunto de pruebas a fin de validar que todos los módulos que componen el sistema actúan y responden según lo esperado:

Prueba N° 1	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Seleccionar ícono de ingreso al sistema dentro del menú Android
<b>Criterio de aprobación:</b>	Ingreso a la aplicación de forma directa sin ningún tipo de publicidad o presentación.

Prueba N° 2	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Acceder al menú de elección de directorios a procesar
<b>Criterio de aprobación:</b>	Acceso al explorador de archivos, pudiendo visualizar el contenido dentro de cada carpeta.

Prueba N° 3	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Verificar estado de procesamiento de las imágenes durante la ejecución.
<b>Criterio de aprobación:</b>	El sistema muestra en todo momento el avance de procesamiento que va obteniendo, a través de métricas generadas sobre cantidad de imágenes procesadas versus cantidad total de imágenes a procesar.

Prueba N° 4	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Verificar informe de finalización de procesamiento de imágenes por parte del sistema.
<b>Criterio de aprobación:</b>	Deberá indicar el estado de finalización (éxito o error). En caso de que haya finalizado exitosamente, a continuación deberá presentar el menú de administración del resultado. Caso contrario, informar el error y reportar el bug.

Prueba N° 5	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Probar las diferentes opciones de administración de imágenes y directorios sugeridos por el resultado.
<b>Criterio de aprobación:</b>	La utilización de dichas opciones debe responder de manera fluida y realizar las tareas especificadas. Debe informar el éxito de cada tarea realizada de manera dinámica y rápida.

Prueba N° 6	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Evaluar satisfacción del usuario
<b>Criterio de aprobación:</b>	Una vez finalizadas las tareas de administración, deberá presentar una pregunta al usuario a fin de conocer el grado de satisfacción del mismo.

Prueba N° 7	
<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Seleccionar el botón de confirmar el resultado final.
<b>Criterio de aprobación:</b>	Las imágenes se deben mover de los directorios originales hacia las nuevas carpetas creadas por la aplicación. Informar al usuario el éxito de la operación.

Prueba N° 8	
-------------	--

<b>Sujeto</b>	Ejecutor del proyecto
<b>Tarea</b>	Seleccionar el botón de eliminar el resultado final.
<b>Criterio de aprobación:</b>	Las imágenes deberán permanecer inalteradas y en el estado original previo a la utilización de la aplicación. Informar al usuario el éxito de la operación.

## 4.5 Recursos para realizar las pruebas del proyecto

### 4.5.1 Pruebas locales

- Notebook Banghó Max 1524 Intel® Core™ i5-3210M CPU @ 2.50GHz × 4, 6gb RAM, 102gb SSD.

### 4.5.2 Pruebas de la aplicación y el motor de procesamiento

- Motorola Moto G4 PLUS XT1641 32GB

## 4.6 Definición de etapas de desarrollo e implementación

A continuación se definirán las condiciones que deberá presentar el desarrollo de toda la aplicación Android y del motor de procesamiento en cada una de sus etapas. Toda la implementación deberá realizarse en 4 etapas, a fin de que la última de ellas devuelva la aplicación completa, integrada y lista para ser utilizable.

- **Prototipo 1: Integración inicial de tecnologías y componentes en desarrollo**

Sistema Android: deberá integrar al motor de procesamiento realizado en C++. La tarea que deberá cumplir es mostrar al usuario un conjunto de directorios con imágenes a agrupar (elegido por defecto), y la selección realizada sea ingresada al motor de procesamiento. Después de que se ejecute el motor, deberá mostrar por pantalla el resultado final del agrupamiento obtenido.

Motor de procesamiento: deberá leer las imágenes ingresadas por la aplicación, y al menos realizar un tipo de procesamiento en cada una de ellas para finalmente emitir un resultado con grupos determinados. No necesariamente se debe aplicar una metodología de clustering (ya que no se espera que esté resuelta), pero sí se espera que el formato de entrada y salida del motor se encuentre lo suficientemente definido.

- **Prototipo 2: Agregación de nuevas características al sistema e implementación de la metodología de clustering en el motor**

Sistema Android: agrega a lo ya realizado la opción de elegir un directorio

determinado. Luego de la ejecución del motor, deberá crear directorios en el dispositivo y mover cada imagen al directorio correspondiente.

Motor de procesamiento: aplicará las metodologías definidas para realizar el procesamiento de las imágenes, devolviéndole al sistema a qué grupo debe ir cada una en base al resultado de clustering obtenido.

- **Prototipo 3: Continuación del desarrollo de nuevas funcionalidades y presentación formal del motor de procesamiento**

Sistema Android: deberá permitir la utilización de las diferentes tareas de administración definidas anteriormente una vez que el procesamiento fue realizado. Además, dará la opción de guardar el resultado final o eliminarlo y volver todo al estado anterior al uso de la misma.

Motor de procesamiento: podrá sufrir modificaciones a fin de lograr mejores resultados. Además, deberá superar con éxito los diferentes criterios de aprobación definidos en el protocolo de experimentación.

- **Prototipo 4: Finalización del desarrollo e integración final de todos los componentes**

Sistema Android: deberá presentar todas las interfaces de usuario terminadas y listas para utilizar, el ícono que distinguirá a la aplicación y la aprobación de todas las pruebas pertenecientes al protocolo de validación.

Motor de procesamiento: podrá sufrir modificaciones a fin de contemplar ciertos aspectos relacionados a la experiencia de usuario.