****

**FACULTAD DE INGENIERÍA**

**CARRERA DE INGENIERÍA DE SISTEMAS**

**COMPUTACIONALES**

**ALL-DETECTOR**

**OpenCv**

**INTEGRANTES:**

* VALVERDE CCASIHUI, JAVIER ENMANUEL
* ARANDA HUARICANCHA, BRYAN
* HINOSTROZA QUILLO, ARMANDO

**CURSO:** VIDEOJUGEOS Y APLICACIONES MOVILES

**CLASE:** 6772

**DOCENTE:** ARROYO TABOADA ÁNGEL DAVID

INDICE

[1. INTRODUCCIÓN 3](#_Toc519391983)

[2. RESUMEN 4](#_Toc519391984)

[3. MARCO TEÓRICO 4](#_Toc519391985)

[4. DIAGRAMA GENERAL DE FLUJOS 6](#_Toc519391986)

[5. DESARROLLO DE APLICACIÓN 7](#_Toc519391987)

[5.1. PROTOTIPO: VALIDAR USUARIO 7](#_Toc519391988)

[5.2. CASO DE USO: VALIDAR USUARIO 7](#_Toc519391989)

[5.3. DIAGRAMA DE CLASES DE ANÁLISIS: VALIDAR USUARIO 8](#_Toc519391990)

[5.4. DIAGRAMA DE ACTIVIDADES: VALIDAR USUARIO 8](#_Toc519391991)

[5.5. PROTOTIPO: REGISTAR USUARIO 9](#_Toc519391992)

[5.6. CASO DE USO: REGISTRAR USUARIO 9](#_Toc519391993)

[5.7. DIAGRAMA DE CLASES DE ANÁLISIS: REGISTAR USUARIO 10](#_Toc519391994)

[5.8. DIAGRAMA DE ACTIVIDADES: REGISTRAR USUARIO 10](#_Toc519391995)

[5.9. PROTOTIPOS: GENERAR DETECCIÓN DE TEXTO 11](#_Toc519391996)

[5.10. CASO DE USO: GENERAR DETECCIÓN DE TEXTO 12](#_Toc519391997)

[5.11. DIAGRAMA DE CLASES DE ANÁLISIS: GENERAR DETECCIÓN DE TEXTO 13](#_Toc519391998)

[5.12. DIAGRAMA DE ACTIVIDADES: GENERAR DETECCIÓN DE TEXTO 13](#_Toc519391999)

[5.13. PROTOTIPOS: GENERAR RECORTE DE IMAGEN 14](#_Toc519392000)

[5.14. CASO DE USO: GENERAR RECORTE DE IMAGEN 14](#_Toc519392001)

[5.15. DIAGRAMA DE CLASES DE ANÁLISIS: GENERAR RECORTE DE IMAGEN 15](#_Toc519392002)

[5.16. DIAGRAMA DE ACTIVIDADES: GENERAR RECORTE DE IMAGEN 15](#_Toc519392003)

[5.17. PROTOTIPO: MEDIR DISTANCIA DE IMAGEN 15](#_Toc519392004)

[5.18. CASO DE USO: MEDIR DISTANCIA DE IMAGEN 16](#_Toc519392005)

[5.19. DIAGRAMA DE CLASES DE ANÁLISIS: MEDIR DISTANCIA DE IMAGEN 17](#_Toc519392006)

[5.20. DIAGRAMA DE ACTIVIDADES: MEDIR DISTANCIA DE IMAGEN 17](#_Toc519392007)

[6. PLAN DE PRUEBAS 18](#_Toc519392008)

[6.1. PLAN DE PRUEBAS VALIDAR USUARIO 18](#_Toc519392009)

[6.2. PLAN DE PRUEBAS REGISTRAR USUARIO. 19](#_Toc519392010)

[6.3. PLAN DE PRUEBAS GENERAR DETECCIÓN DE TEXTO 19](#_Toc519392011)

[6.4. PLAN DE PRUEBAS GENERAR RECORTE DE IMAGEN 20](#_Toc519392012)

[6.5. PLAN DE PRUEBAS MEDIR DISTANCIA DE IMAGEN 21](#_Toc519392013)

# INTRODUCCIÓN

En la actualidad existen muchas tecnologías que están revolucionando el mercado de software. Una de las de mayor impacto son los aplicativos móviles. En la cual podremos encontrar herramientas útiles desde la flexibilidad de un smartphone. Asimismo, esto es posible en parte gracias a sus potentes sistemas operativos. Para resaltar, Android, que usaremos en nuestro proyecto, es uno de los más populares dentro del mercado, gracias a su flexibilidad, repositorios, potencia entre otros nos permite a través de una de sus plataformas Android Studio desarrollar una gran variedad de proyectos.

En este proyecto nuestro grupo asumo el reto de desarrollar un aplicativo móvil en Android, utilizando las librerías de OPENCV. La cual nos brindara las herramientas necesarias para su realización.

En este documento detallaremos las especificaciones del aplicativo, así como sus casos de uso, diagramas y plan de pruebas.

# RESUMEN

Nuestro proyecto tiene por nombre ALL-DETECTOR.

Es un aplicativo móvil desarrollado en Android, que nos permite medir la distancia entre el dispositivo y una persona previamente definida. Por otro lado, también cuenta con un detector de texto en imagen, que extraerá el texto y lo llevará a una imagen independiente, también permite realizar un recorte de una imagen donde se encuentre una persona y cambiar a otro fondo como se decida. Para ello, el cliente deberá estar registrado en la aplicación. Por ende, habrá una validación de datos para iniciar la herramienta. Todo esto nos permite realizar la librería OPENCV, el cual cuenta con algoritmos muy complejos que nos ayudaran a realizar las validaciones y registro de imágenes.

ALL-DETECTOR nos permite tener herramientas muy útiles en un solo aplicativo. Esto beneficia la utilización de recursos del dispositivo y la comodidad de tener estas especificaciones en un mismo lugar.

# MARCO TEÓRICO

**Opencv.**

Es una librería de software libre para todo lo relacionado a visión artificial.

Cuenta con mas de 2500 algoritmos. Estos te permiten identificar un objeto predefinido, extraer modelos 3D, reconocer escenarios entre otros.

**Android.\_**

Es un sistema operativo basado en Linux, diseñado en especial para plataforma móvil. Inicialmente desarrolado por Android INC. Es uno de los sistemas operativos con mayor alcance y confianza para el desarrollo de potentes aplicativos.

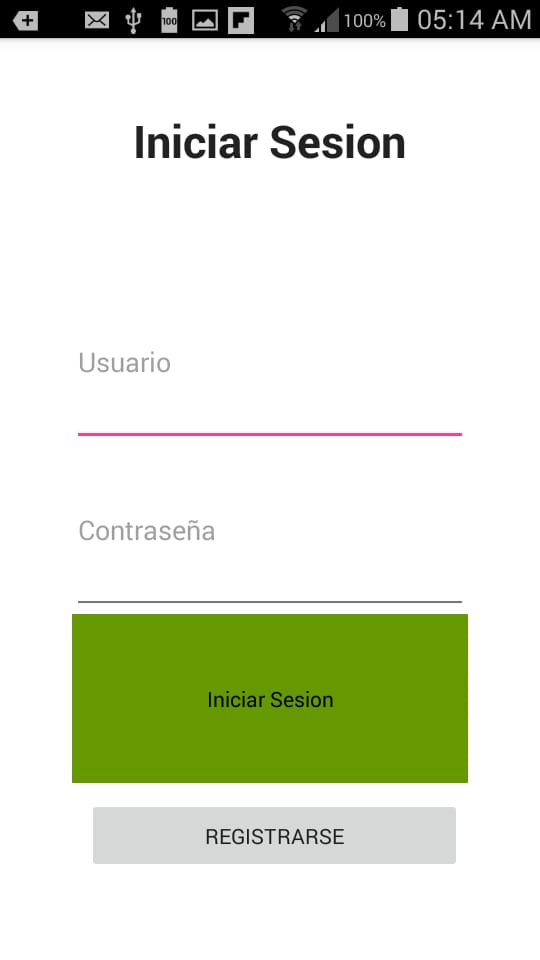
**Android Studio.\_**

Es un entorno de desarrollo para realizar aplicativos móviles. Ideal para aplicaciones para Android.

# DIAGRAMA GENERAL DE FLUJOS

Figura 1: Diagrama General de Flujos

# 5. DESARROLLO DE APLICACIÓN

5.1. PROTOTIPO: VALIDAR USUARIO  
Figura 2: Prototipo validar usuario Figura 3: Prototipo menú principal

# 5.2. CASO DE USO: VALIDAR USUARIO

**1. Breve descripción**

El caso de uso permite al usuario ingresar a su cuenta validando sus datos para poder ingresar a la aplicación.

**2. Actor(es)**

Usuario

**3. Flujo de Eventos**

**3.1 Evento disparador**

El caso de uso comienza cuando el usuario ingresa a la aplicación.

**3.2. Flujo Básico.**

1. El sistema muestra la interfaz “Validar usuario” con los siguientes campos:

Datos del usuario: Usuario, contraseña.  
Además, las opciones: Iniciar sesión y Registrarse.

1. El usuario ingresa cuenta de usuario.
2. El usuario ingresa contraseña.
3. El usuario selecciona la opción Iniciar sesión.
4. El sistema valida los datos del usuario.
5. El sistema muestra la interfaz “Menú Principal” y finaliza el caso de uso.

**3.3. Sub Flujo**

No aplica.

**3.4. Flujos Alternativos**

**<Validar Usuario >**

Si en el punto 5, el sistema detecta que la cuenta y contraseña ingresada son incorrectas, muestra un mensaje “Datos Incorrectos”.

**<Restringir Acceso >**

Si en el punto 5, el sistema detecta que por tercera vez los datos ingresados fueron incorrectos, inhabilita la opción Iniciar sesión.

**4. Pre Condiciones**

1. El Usuario debe estar registrado en el sistema.

**5. Post Condiciones**

No aplica.

**6. Puntos de extensión**

Extiende al caso de uso “Registrar Usuario”.

**7. Requerimientos Especiales**  No aplica.

# 5.3. DIAGRAMA DE CLASES DE ANÁLISIS: VALIDAR USUARIO



Figura 4: Diagrama de clases de análisis validar usuario

# 5.4. DIAGRAMA DE ACTIVIDADES: VALIDAR USUARIO



Figura 5: Diagrama de actividades validar usuario

# 5.5. PROTOTIPO: REGISTAR USUARIO

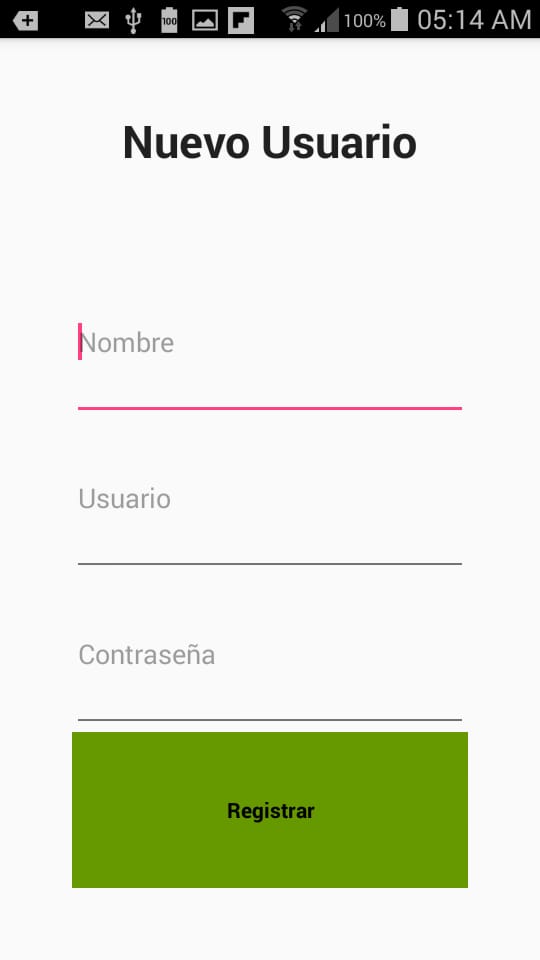


Figura 6: Prototipo registrar usuario

# 5.6. CASO DE USO: REGISTRAR USUARIO

**1. Breve descripción**

El caso de uso permite al usuario registrar sus datos en una cuenta nueva para acceder a la aplicación.

**2. Actor(es)**

Usuario

**3. Flujo de Eventos**

**3.1 Evento disparador**

El Caso de uso se inicia cuando el usuario selecciona la opción “Registrarse” en la interfaz Validar Usuario.

**3.2. Flujo Básico.**

1. El sistema muestra la interfaz “Registrar Usuario” con los siguientes campos:

Datos del usuario: Nombre, usuario y contraseña.  
Además, la opción: Registrar.

1. El usuario ingresa Nombre.
2. El usuario ingresa Usuario.
3. El usuario ingresa Contraseña.
4. El usuario selecciona la opción Registrar.
5. El sistema valida los datos.
6. El sistema guarda los datos del usuario y muestra un mensaje “Registrado”
7. El sistema regresa al caso de uso base y finaliza el caso de uso.

**3.3. Sub Flujo**

No aplica.

**3.4. Flujos Alternativos**

**<Validar datos >**

Si en el punto 6, el sistema detecta que se ha ingresado un usuario repetido, muestra un mensaje “El usuario ya existe”.

**4. Pre Condiciones**

Listado de usuario disponible

**5. Post Condiciones**

El sistema registra los datos del nuevo usuario.

**6. Puntos de extensión**

No aplica.

**7. Requerimientos Especiales**  No aplica.

# 5.7. DIAGRAMA DE CLASES DE ANÁLISIS: REGISTAR USUARIO



Figura 7: Diagrama de clases de análisis registrar usuario

# 5.8. DIAGRAMA DE ACTIVIDADES: REGISTRAR USUARIO



Figura 8: Diagrama de actividades registrar usuario

# 5.9. PROTOTIPOS: GENERAR DETECCIÓN DE TEXTO

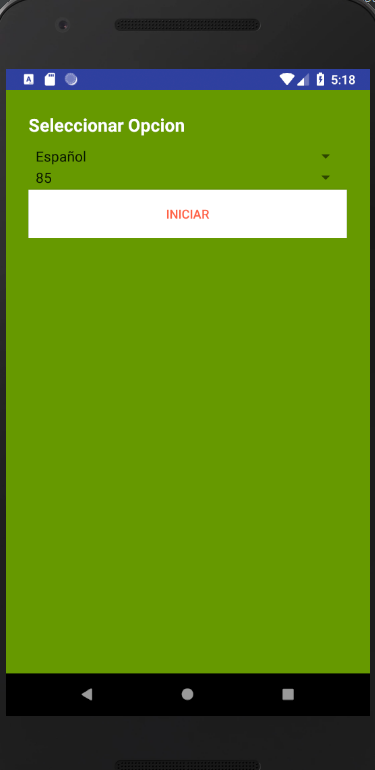


Figura 9: Prototipo configurar detección Figura 10: Prototipo llamar cámara

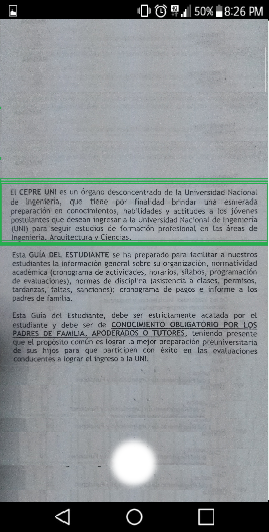
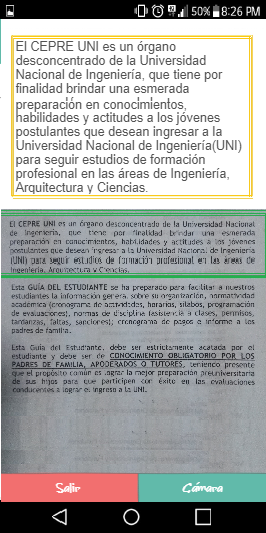


Figura 11: Prototipo seleccionar recorte Figura 12: Prototipo detección de texto

# 5.10. CASO DE USO: GENERAR DETECCIÓN DE TEXTO

**1. Breve descripción**

El caso de uso permite al usuario reconocer el texto de una imagen fotografiada con la cámara del dispositivo.

**2. Actor(es)**

Usuario

**3. Flujo de Eventos**

**3.1 Evento disparador**

El caso de uso comienza cuando el usuario selecciona “Reconocimiento de texto” en el menú principal.

**3.2. Flujo Básico.**

1. El sistema muestra la interfaz “Configurar detección” con dos listas combinadas: Idioma y Calidad. Además, la opción Iniciar.
2. El usuario selecciona la calidad de detección de texto.
3. El usuario selecciona la opción Iniciar.
4. El sistema muestra la interfaz “Llamar cámara”
5. El usuario selecciona la opción cámara.
6. El sistema muestra la interfaz “Seleccionar Recorte”.
7. El usuario captura la foto de un texto.
8. El sistema muestra la interfaz "Detección de Texto" donde se muestra el texto detectado. Además, incluye los botones: Cámara y Salir.
9. El usuario selecciona la opción Salir, el sistema muestra el menú principal y se finaliza el caso de uso.

**3.3. Sub Flujo**

No aplica.

**3.4. Flujos Alternativos**

**<Seleccionar texto>**

Si en el punto 7, el usuario no ha seleccionado un texto para la captura, el sistema muestra un mensaje “Capture un texto”

**<Tomar nueva foto>**

Si en el punto 8, el usuario desea tomar una nueva foto.

El usuario selecciona la opción Cámara, el sistema muestra la interfaz “Seleccionar Recorte”.

**4. Pre Condiciones**

1. El Usuario debe estar logueado en el sistema.

**5. Post Condiciones**

No aplica

**6. Puntos de extensión**

No aplica.

**7. Requerimientos Especiales** No aplica.

# 5.11. DIAGRAMA DE CLASES DE ANÁLISIS: GENERAR DETECCIÓN DE TEXTO

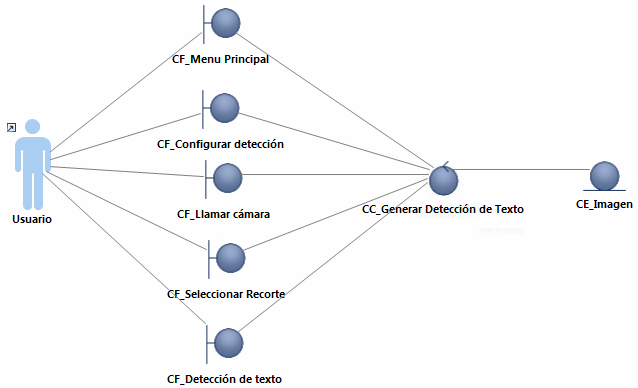


Figura 13: Diagrama de clases de análisis generar detección de texto

# 5.12. DIAGRAMA DE ACTIVIDADES: GENERAR DETECCIÓN DE TEXTO

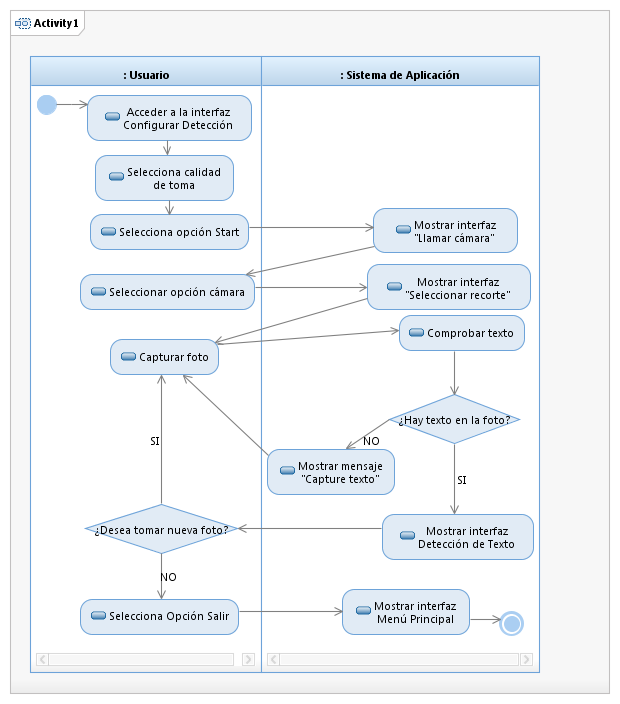


Figura 14: Diagrama de actividades generar detección de texto

# F:\VIDEO JUEGOS\APLICACIONES MOVILES _FINAL\DISEÑOS_FINAL_MOVILES\Deteccion y Recorte\WhatsApp Image 2018-07-14 at 5.23.57 AM.jpegF:\VIDEO JUEGOS\APLICACIONES MOVILES _FINAL\DISEÑOS_FINAL_MOVILES\Deteccion y Recorte\1.jpeg 5.13. PROTOTIPOS: GENERAR RECORTE DE IMAGEN

Figura 15: Prototipo seleccionar vista Figura 16: Prototipo capturar recorte

# 5.14. CASO DE USO: GENERAR RECORTE DE IMAGEN

**1. Breve descripción**

El caso de uso permite al usuario recortar la silueta de una persona fotografiada con la cámara del dispositivo para luego mostrarlo en la parte superior derecha de la pantalla.

**2. Actor(es)**

Usuario

**3. Flujo de Eventos**

**3.1 Evento disparador**

El caso de uso comienza cuando el usuario selecciona “Recorte de imagen” en el menú principal.

**3.2. Flujo Básico.**

1. El sistema muestra la Interfaz “SELECCIONAR VISTA”. Además, muestra las opciones: COLOR GRIS y COLOR RGB.

2. El usuario selecciona la opción COLOR RGB.

3. El sistema muestra la interfaz “CAPTURAR RECORTE”.

4. El usuario enfoca a una persona.

5. El sistema detecta a través de un cuadro a la persona.

6. El sistema muestra el recorte de imagen de la persona enfocada en la parte superior derecha.

7. Si la persona enfocada cambia de gesto.

8. El sistema agrega el recorte de imagen de la persona actualizada y finaliza el caso de uso.

**3.3. Sub Flujo**

No aplica.

**3.4. Flujos Alternativos**

**<Tomar nueva foto>**

Si en el punto 2, el usuario selecciona la opción Color GRIS, el sistema muestra la interfaz “CAPTURAR RECORTE” en formato gris.

**4. Pre Condiciones**

1. El Usuario debe estar logueado en el sistema.

2. Debe haber una persona en el enfoque de la cámara para que sea detectable.

**5. Post Condiciones**

No aplica.

**6. Puntos de extensión**

No aplica.

**7. Requerimientos Especiales** No aplica.

# 5.15. DIAGRAMA DE CLASES DE ANÁLISIS: GENERAR RECORTE DE IMAGEN

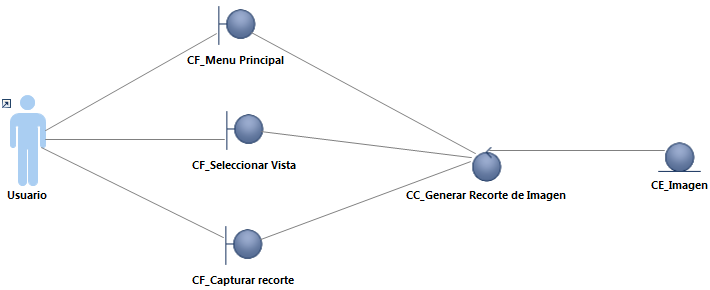


Figura 17: Diagrama de clases de análisis generar recorte de imagen

# 5.16. DIAGRAMA DE ACTIVIDADES: GENERAR RECORTE DE IMAGEN

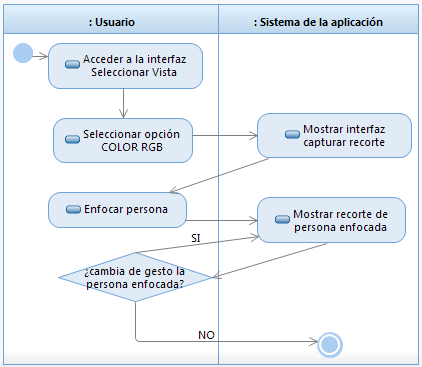


Figura 18: Diagrama de actividades generar recorte de imagen

# 5.17. PROTOTIPO: MEDIR DISTANCIA DE IMAGEN

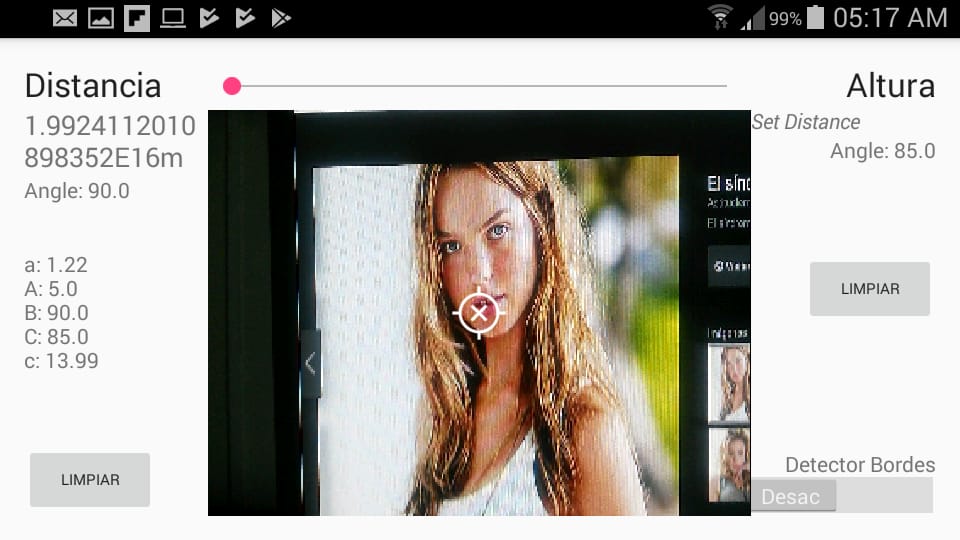


Figura 19: Prototipo Detector de distancia

# 5.18. CASO DE USO: MEDIR DISTANCIA DE IMAGEN

**1. Breve descripción**

El caso de uso permite al usuario mediar la distancia de una persona hacia la cámara del usuario

**2. Actor(es)**

Usuario

**3. Flujo de Eventos**

**3.1 Evento disparador**

El caso de uso comienza cuando el usuario selecciona “Detector de distancia” en el menú principal.

**3.2. Flujo Básico.**

1. El sistema muestra la interfaz “DETECTOR DE DISTANCIA” con el campo Distancia y Altura. Además, incluye las opciones de limpiar y una opción de estado de Detector de Bordes.
2. El usuario enfoca a una persona.
3. El sistema detecta a la persona y lo marca a través de un punto.
4. El sistema muestra la distancia entre la persona y la cámara.
5. Si el usuario mueve la cámara o la persona detectada se mueve.
6. El sistema actualiza la distancia entre la persona y la cámara y finaliza el caso de uso.

**3.4. Sub Flujo**

No aplica.

**3.3. Flujos Alternativos**

**<buscar persona>**

Si en el punto 4, el sistema no detecta la forma predefinida de una persona no podrá mostrar la distancia.

**<Detectar bordes>**

Si en el punto 2, el usuario activa la opción Detector de Bordes, el sistema muestra la interfaz griseada.

**4. Pre Condiciones**

1. El Usuario debe estar logueado en el sistema.

2. Debe haber una persona en el enfoque de la cámara para que sea detectable.

**5. Post Condiciones**

No aplica

**6. Puntos de extensión**

No aplica.

**7. Requerimientos Especiales** No aplica.

# 5.19. DIAGRAMA DE CLASES DE ANÁLISIS: MEDIR DISTANCIA DE IMAGEN

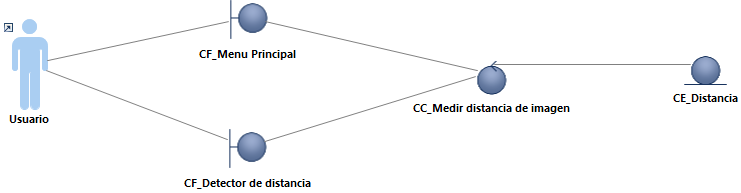


Figura 20: Diagrama de clases de medir distancia de imagen

# 5.20. DIAGRAMA DE ACTIVIDADES: MEDIR DISTANCIA DE IMAGEN

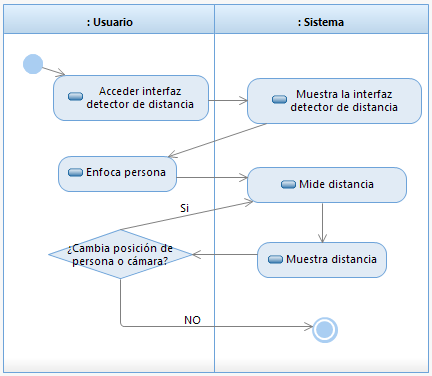


Figura 21: Diagrama de actividades medir distancia de imagen

# 6. PLAN DE PRUEBAS

# PLAN DE PRUEBAS VALIDAR USUARIO

|  |  |
| --- | --- |
| PRUEBA 1 | |
| Tarea | Ingresar al Sistema |
| Descripción | El sistema valida si las credenciales del usuario son las registradas en la base de datos. |
| Realización | El usuario ingresa los parámetros necesarios, el sistema realizará una validación de los datos ingresados consultando en la base de datos. |
| Resultado esperado | EL usuario acceda al sistema. |
| Resultado obtenido | El usuario ingresa satisfactoriamente al sistema, el sistema muestra el menú principal de la aplicación. |
| Posible condición de fallo | Los parámetros ingresados no coinciden con los datos de la tabla usuario. |
| Estado Corrección | 100% |

|  |  |
| --- | --- |
| PRUEBA 2 | |
| Tarea | Restringir acceso al Sistema |
| Descripción | El sistema valida si las credenciales del usuario son las registradas en la base de datos. |
| Realización | El usuario ingresa mal los parámetros por tercera vez, el sistema restringirá el acceso al menú principal de la aplicación. |
| Resultado esperado | EL usuario no puede acceder al sistema. |
| Resultado obtenido | El sistema inhabilita la opción Iniciar sesión por seguridad, el usuario no podrá acceder al sistema. |
| Posible condición de fallo | El usuario reinicia la aplicación. |
| Estado Corrección | 100% |

# PLAN DE PRUEBAS REGISTRAR USUARIO.

|  |  |
| --- | --- |
| PRUEBA 3 | |
| Tarea | Registrar Usuario |
| Descripción | EL sistema permite que un usuario pueda registrarse en el sistema. |
| Realización | El sistema obtiene los parámetros ingresados por el usuario y lo registra en la base de datos. |
| Resultado esperado | El usuario queda registrado en la base de datos del sistema. |
| Resultado obtenido | El usuario se registró correctamente en el sistema. |
| Posible condición de fallo | El parámetro usuario existe en la base de datos del sistema. |
| Estado Corrección | 100% |

# PLAN DE PRUEBAS GENERAR DETECCIÓN DE TEXTO

|  |  |
| --- | --- |
| PRUEBA 4 | |
| Tarea | Detectar texto en pantalla |
| Descripción | EL sistema permite detectar un texto de una imagen a través de un cuadro de detección de la pantalla. |
| Realización | El usuario enfoca un texto por medio de un cuadro de detección de la aplicación. |
| Resultado esperado | El usuario puede observar con claridad el texto detectado en la parte superior de la siguiente pantalla. |
| Resultado obtenido | El sistema detecta correctamente el texto de una imagen y lo muestra en la parte superior de la siguiente pantalla. |
| Posible condición de fallo | Texto de imagen borroso. |
| Estado Corrección | 100% |

# PLAN DE PRUEBAS GENERAR RECORTE DE IMAGEN

|  |  |
| --- | --- |
| PRUEBA 5 | |
| Tarea | Recortar rostro de persona |
| Descripción | EL sistema permite detectar el rostro de una persona a través de un cuadro de detección de la pantalla. |
| Realización | El usuario enfoca una persona por medio de un cuadro de detección de la aplicación. |
| Resultado esperado | El usuario puede observar el recorte de rostro de la persona enfocada. |
| Resultado obtenido | El sistema detecta correctamente el rostro de una persona y lo muestra en la parte superior derecha de la pantalla. |
| Posible condición de fallo | No enfocar persona. |
| Estado Corrección | 100% |

|  |  |
| --- | --- |
| PRUEBA 6 | |
| Tarea | Capturar gesto de persona |
| Descripción | EL sistema permite detectar el gesto de una persona a través de un cuadro de detección de la pantalla. |
| Realización | El usuario enfoca una persona por medio de un cuadro de detección de la aplicación. |
| Resultado esperado | El usuario puede observar la agregación del recorte del gesto de la persona enfocada. |
| Resultado obtenido | El sistema detecta correctamente el gesto de una persona y lo agrega en la parte inferior del recorte anterior. |
| Posible condición de fallo | Persona detectada no cambia de gesto. |
| Estado Corrección | 100% |

# PLAN DE PRUEBAS MEDIR DISTANCIA DE IMAGEN

|  |  |
| --- | --- |
| PRUEBA 7 | |
| Tarea | Detectar distancia de persona |
| Descripción | EL sistema permite medir la distancia que existe entre la persona enfocada y la cámara del dispositivo. |
| Realización | El usuario enfoca una persona por medio de un círculo de detección de la aplicación. |
| Resultado esperado | El usuario puede observar la distancia que existe entre la cámara del dispositivo y la persona enfocada. |
| Resultado obtenido | El sistema calcula correctamente la distancia que existe entre la cámara del dispositivo y la persona enfocada. |
| Posible condición de fallo | No enfocar persona. |
| Estado Corrección | 100% |

|  |  |
| --- | --- |
| PRUEBA 8 | |
| Tarea | Actualizar distancia de persona |
| Descripción | EL sistema permite actualizar la distancia que existe entre la persona enfocada y la cámara del dispositivo. |
| Realización | El usuario mueve la cámara o la persona enfocada se mueve. |
| Resultado esperado | El usuario puede observar la actualización de distancia que existe entre la cámara del dispositivo y la persona enfocada. |
| Resultado obtenido | El sistema calcula correctamente la distancia que existe entre la cámara del dispositivo y la persona enfocada. |
| Posible condición de fallo | No enfocar persona. |
| Estado Corrección | 100% |