



## **DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

### **CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA**

#### **TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: IMPLEMENTACIÓN DE UN SISTEMA DE SOPORTE EN  
LÍNEA CON RECONOCIMIENTO FACIAL COMO MECANISMO DE  
SEGURIDAD PARA REPRESENTANTES DE PFIZER**

#### **AUTORES:**

**PILLAJO COKA, JONATHAN ANDRÉS  
YAGUANA MONTERO, JEFFERSON ESTEBAN**

**DIRECTOR: ING. GUERRERO IDROVO, ROSA GRACIELA**

**SANGOLQUÍ**

**2018**

## CERTIFICADO



VICERRECTORADO ACADEMICO GENERAL  
SISTEMA INTEGRADO DE BIBLIOTECAS SIB-ESPE



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN  
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

### CERTIFICACIÓN

Certifico que el trabajo de titulación, "*Implementación de un sistema de soporte en línea con reconocimiento facial como mecanismo de seguridad para representantes de Pfizer*" fue realizado por los señores *Pillajo Coka, Jonathan Andrés y Yaguana Montero, Jefferson Esteban*, el mismo que ha sido revisado en su totalidad, analizado por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 28 de Febrero del 2018

Firma:

Ing. Rosa Graciela Guerrero Idrovo

C. C. 1720513322

## AUTORIA DE RESPONSABILIDAD



VICERRECTORADO ACADEMICO GENERAL  
SISTEMA INTEGRADO DE BIBLIOTECAS SIB-ESPE



### DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

#### AUTORÍA DE RESPONSABILIDAD

Nosotros, **Pillajo Coka, Jonathan Andrés y Yaguana Montero, Jefferson Esteban**, declaramos que el contenido, ideas y criterios del trabajo de titulación: **Título: Implementación de un sistema de soporte en línea con reconocimiento facial como mecanismo de seguridad para representantes de Pfizer**, es de mi autoría y responsabilidad, cumpliendo con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de la investigación mencionada es veraz.

Sangolquí, 2 de Marzo del 2018

Firma

Jonathan Andrés Pillajo Coka

C.C.: 1715824775

Firma

Jefferson Esteban Yaguana Montero

C.C.: 172287113-9

## **AUTORIZACIÓN**



VICERRECTORADO ACADEMICO GENERAL  
SISTEMA INTEGRADO DE BIBLIOTECAS SIB-ESPE



**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN  
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA**

### **AUTORIZACIÓN**

Nosotros, *Pillajo Coka, Jonathan Andrés y Yaguana Montero, Jefferson Esteban*, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Título: Implementación de un sistema de soporte en línea con reconocimiento facial como mecanismo de seguridad para representantes de Pfizer en el Repositorio Institucional**, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Sangolquí, 2 de Marzo del 2018

Firma

**Jonathan Andrés Pillajo Coka**

C.C.: *1115824773*

Firma

**Jefferson Esteban Yaguana Montero**

C.C.: *172287713-9*

## **DEDICATORIA**

A mis padres, Alicia Montero y Rolan Yaguana, quienes me apoyaron en todo momento, supieron aconsejarme y alentarme en todo este camino para poder lograr esta meta.

A mis hermanos, Oscar y Nicole Yaguana, y a mi sobrino Eduardo López, quienes fueron inspiración para poder lograr esta meta.

A mis Abuelitos, Fanny Sánchez, Ulbio Montero, María Padilla y Justo Yaguana, quienes supieron criarme con los valores que se necesitan para poder lograr las metas que uno se propone.

A mi novia Karla Muñoz quien ha estado a mi lado apoyándome en todo momento para lograr cada una de las metas que me he propuesto y me ha dado su amor incondicional.

A mis tíos Francisco Yaguana y Martha Erazo quienes desde pequeño me han sabido aconsejar y darme su apoyo en cualquier situación en la que me he encontrado, y a pesar de la distancia me han sabido apoyar.

A todos mis familiares quienes estuvieron ahí apoyándome en este largo camino con sus palabras de aliento, estando a mi lado cuando los necesite y brindándome todos los consejos que me han dado.

A mis amigos, con los cuales a pesar de todas las adversidades que se nos han presentado siempre estuvieron ahí dándome su apoyo y sabiendo dar un consejo en cada circunstancia que lo necesite.

Jefferson Esteban Yaguana Montero

## **DEDICATORIA**

Dedico este trabajo de titulación, en primer lugar, a Dios, por haberme dado la oportunidad de llegar a este momento tan importante de mi vida profesional, siendo una guía en los triunfos y ayudándome a superar los momentos difíciles.

A mi madre, Janeth Coka, por haberme dado la vida y siempre estar presente en ella, brindándome su apoyo incondicional. Por haberme enseñado lo que está bien y lo que no, para así tomar mis propias decisiones, aunque a veces parezcan incorrectas. Y por siempre demostrar lo orgullosa que se siente tanto de mí como de mi hermano.

A mi hermano, Alejandro Pillajo, que, aunque no seamos muy unidos, sabemos que podemos contar el uno con el otro siempre. Por esas horas que pasamos jugando FIFA para aliviar el estrés, a pesar de que se enoja si le gano.

A toda mi familia, abuelos, tíos, primos, por el apoyo y la preocupación que siempre demuestran hacia mi persona. Y por los consejos que me supieron dar para mi vida.

A mis amigos, por las farras, las conversaciones, los apodos, las idas al bloque A, y por todo lo que hizo que las clases sean más llevaderas. Por el apoyo y acolite en los momentos de locura.

Jonathan Andrés Pillajo Coka

## **AGRADECIMIENTO**

En primer lugar, me gustaría agradecer a Dios por permitirme vivir cada día con las personas que quiero, mi familia, mi novia, mis amigos y en paz conmigo mismo.

Seguido quiero agradecer a la Universidad de la Fuerzas Armadas “ESPE” por permitirme educarme en sus aulas y ayudarme a desarrollar mis habilidades profesionales y poder vivir experiencias extraordinarias.

A mi directora de tesis Ingeniera Graciela Guerrero, por brindarnos su apoyo incondicional en todo momento, y brindarnos su tiempo para poder desarrollar este proyecto de investigación.

A la empresa Pfizer por brindarnos su apertura para desarrollar este proyecto de investigación, brindarnos todo su apoyo y los recursos necesarios para salir adelante con este proyecto.

A todos los ingenieros quienes nos inculcaron su enseñanza y los valores que se necesitan para poder desarrollarse en el ámbito profesional, y poder salir adelante dejando en alto el nombre de la Universidad de las Fuerzas Armadas “ESPE”.

Para finalizar quiero agradecer a mi familia, novia y amigos que estuvieron apoyándonos y permitieron que sea posible la culminación de este proyecto de investigación.

Jefferson Esteban Yaguana Montero

## **AGRADECIMIENTO**

Agradezco a Dios por las decisiones que tomé, que bien o mal han sido para mejorar y me enseñaron a superar obstáculos.

A mi madre, hermano, a toda mi familia y amigos, por el apoyo constante que me demostraron para alcanzar esta meta.

A la Universidad de las Fuerzas Armadas, que ha sido como un segundo hogar durante los últimos años, en donde viví muchas experiencias tanto alegres como no tan felices, pero que me ayudaron a convertirme en la persona que soy.

A la Ingeniera Graciela Guerrero, directora de tesis, por su motivación y excelente asesoramiento en la realización de este trabajo.

A Pfizer por abrirnos sus puertas y brindarnos su apoyo para realizar este trabajo de titulación.

Por último, quiero agradecer a esa persona que conocí el día que egresé, por ser mi amiga y permitirme ser su amigo, por contarme sus historias y escuchar las mías, por las risas y por enseñarme a encontrarle lo divertido a la vida.

Jonathan Andrés Pillajo Coka

## ÍNDICE

<b>CERTIFICADO .....</b>	ii
<b>AUTORIA DE RESPONSABILIDAD .....</b>	iii
<b>AUTORIZACIÓN.....</b>	iv
<b>DEDICATORIA .....</b>	v
<b>DEDICATORIA .....</b>	vi
<b>AGRADECIMIENTO .....</b>	vii
<b>AGRADECIMIENTO .....</b>	viii
<b>ÍNDICE .....</b>	ix
<b>ÍNDICE DE TABLAS.....</b>	xii
<b>ÍNDICE DE FIGURAS .....</b>	xiii
<b>RESUMEN .....</b>	xv
<b>ABSTRACT .....</b>	xvi
<b>CAPÍTULO I INTRODUCCIÓN .....</b>	1
1.1. Antecedentes.....	1
1.2. Planteamiento del problema .....	2
1.3. Justificación .....	2
1.4. Objetivos.....	3
1.4.1. Objetivo General .....	3
1.4.2. Objetivos Específicos.....	3
1.5. Alcance .....	3
1.6. Hipótesis .....	4
1.7. Estructura del Trabajo de Investigación .....	4
<b>CAPÍTULO II TRABAJOS RELACIONADOS Y ANÁLISIS DE TÉCNICAS Y ALGORITMOS DE RECONOCIMIENTO FACIAL .....</b>	6
2.1. Técnicas de Reconocimiento Facial .....	6
2.1.1. Análisis de Componentes Principales (PCA) .....	11
2.1.2. Análisis de Discriminante Lineal (LDA) .....	12
2.1.3. Transformada de Coseno Discreta (DCT) .....	13
2.1.4. Proyecciones Locales de Preservación (LPP) .....	13
2.1.5. Mapeo de Características Isométricas (ISOMAP) .....	14
2.2. Análisis de algoritmos de Reconocimiento Facial.....	15
2.2.1. PCA - Análisis de Componentes Principales (Eigenfaces).....	15

2.2.2. LDA - Análisis de Discriminante Lineal (Fisherfaces) .....	17
2.2.3. DCT - Transformada de Coseno Discreta.....	20
2.2.4. LPP - Proyecciones Locales de Preservación .....	21
2.3. Selección de método de reconocimiento facial para la implementación en el sistema de soporte .....	23
<b>CAPÍTULO III IMPLEMENTACIÓN DEL RECONOCIMIENTO FACIAL.....</b>	<b>27</b>
3.1. Enfoque PCA para Reconocimiento Facial .....	27
3.2. Análisis de Componentes Principales (Eigenfaces) .....	28
3.3. Arquitectura del sistema de reconocimiento facial.....	31
3.4. Implementación del algoritmo PCA en JAVA .....	33
3.4.1. Preparación de la imagen .....	33
3.4.2. Creación del vector imagen.....	34
3.4.3. Obtención de eigenvectors y eigenvalues .....	34
3.5. Ejecución del sistema web de reconocimiento facial .....	36
3.5.1. Guardar rostro (entrenamiento).....	36
3.5.2. Reconocer un rostro nuevo .....	38
<b>CAPÍTULO IV DESARROLLO E IMPLEMENTACION DEL SISTEMA DE SOPORTE EN LÍNEA.....</b>	<b>40</b>
4.1. Metodología SCRUM.....	40
4.2. Requerimientos Funcionales y No Funcionales .....	43
4.3. Historias de Usuarios .....	44
4.4. Modelo físico de base de datos.....	62
4.5. Diagramas de Casos de Uso .....	63
4.5.1. Definición de Actores del Sistema.....	63
4.5.2. Definición de tareas por Actor .....	63
4.5.3. Definición de casos de uso general .....	64
4.5.4. Descripción de casos de uso.....	64
4.6. Diagramas de Secuencia.....	70
4.6.1. Diagrama de Secuencia “Autentificación de Usuario” .....	70
4.6.2. Diagrama de Secuencia “Realizar Solicitud” .....	71
4.6.3. Diagrama de Secuencia “Seguimiento Solicitud”.....	72
4.6.4. Diagrama de Secuencia “Histórico Solicitud” .....	73
4.6.5. Diagrama de Secuencia “Responder Solicitud” .....	73
4.6.6. Diagrama de Secuencia “Escalar Solicitud” .....	74

4.6.7. Diagrama de Secuencia “Histórico Solicitud Pendiente” .....	74
4.7. Diagramas de Navegación .....	75
4.7.1. Diagrama de Navegación “Usuario Administrador” .....	75
4.7.2. Diagrama de Navegación “Usuario Representante” .....	75
4.7.3. Diagrama de Navegación “Usuario Soporte” .....	76
4.8. Arquitectura del sistema .....	76
<b>CAPÍTULO V PRUEBAS Y ANÁLISIS DE RESULTADOS</b> .....	78
5.1. Escenario de prueba.....	78
5.1.1. Paso 1: Utilización del sistema de reconocimiento facial para entrenamiento .....	78
5.1.2. Paso 2: Utilización del sistema de reconocimiento facial para reconocimiento .....	80
5.1.3. Paso 3: Encuesta a los sujetos de prueba .....	80
5.2. Tabulación de los resultados encuesta.....	81
5.3. Análisis de Resultados .....	89
5.3.1. Solución a los problemas detectados .....	91
5.4. Carta de conformidad del proyecto AUTOMATZER. ....	93
<b>CAPÍTULO VI CONCLUSIONES Y RECOMENDACIONES</b> .....	94
6.1. CONCLUSIONES .....	94
6.1. RECOMENDACIONES .....	94
<b>REFERENCIAS</b> .....	96

## ÍNDICE DE TABLAS

<b>Tabla 1</b> Historia del reconocimiento facial .....	6
<b>Tabla 2</b> Ventajas e Inconvenientes de las técnicas basadas en apariencia .....	10
<b>Tabla 3</b> Escala de complejidad matemática .....	24
<b>Tabla 4</b> Escala de entrenamiento previo .....	24
<b>Tabla 5</b> Escala de base de datos .....	24
<b>Tabla 6</b> Escala de procesamiento y toma de decisión .....	25
<b>Tabla 7</b> Requerimientos Funcionales .....	43
<b>Tabla 8</b> Requerimientos No Funcionales .....	44
<b>Tabla 9</b> Historia de Usuario HUPA001 .....	45
<b>Tabla 10</b> Historia de Usuario HUPA002 .....	48
<b>Tabla 11</b> Historia de Usuario HUPA003 .....	50
<b>Tabla 12</b> Historia de Usuario HUPA004 .....	52
<b>Tabla 13</b> Historia de Usuario HUPA005 .....	54
<b>Tabla 14</b> Historia de Usuario HUPA006 .....	56
<b>Tabla 15</b> Historia de Usuario HUPA007 .....	59
<b>Tabla 16</b> Caso de Uso CU_001 “Autentificarse” .....	64
<b>Tabla 17</b> Caso de Uso CU_002 “Realizar Solicitud” .....	65
<b>Tabla 18</b> Caso de Uso CU_003 “Seguimiento Solicitud” .....	66
<b>Tabla 19</b> Caso de Uso CU_004 “Histórico Solicitud” .....	67
<b>Tabla 20</b> Caso de Uso CU_005 “Responder Solicitud” .....	68
<b>Tabla 21</b> Caso de Uso CU_006 “Escalar Solicitud” .....	69
<b>Tabla 22</b> Caso de Uso CU_007 “Histórico Solicitud Pendiente” .....	70
<b>Tabla 23</b> Encuesta: Pregunta 1 .....	81
<b>Tabla 24</b> Encuesta: Pregunta 2 .....	81
<b>Tabla 25</b> Encuesta: Pregunta 3 .....	82
<b>Tabla 26</b> Encuesta: Pregunta 4 .....	83
<b>Tabla 27</b> Encuesta: Pregunta 5 .....	83
<b>Tabla 28</b> Encuesta: Pregunta 6 .....	84
<b>Tabla 29</b> Encuesta: Pregunta 7 .....	85
<b>Tabla 30</b> Encuesta: Pregunta 8 .....	86
<b>Tabla 31</b> Encuesta: Pregunta 9 .....	87
<b>Tabla 32</b> Encuesta: Pregunta 10 .....	88
<b>Tabla 33</b> Resultados segunda prueba .....	92

## ÍNDICE DE FIGURAS

<b>Figura 1</b> Técnicas de reconocimiento facial.....	9
<b>Figura 2</b> Diagrama de bloques .....	11
<b>Figura 3</b> Componentes principales de conjunto de puntos bidimensional .....	12
<b>Figura 4</b> Separabilidad de las muestras proyectadas en dos dimensiones.....	13
<b>Figura 5</b> Ejemplo de Isomap .....	15
<b>Figura 6</b> Reducción dimensional al aplicar PCA .....	17
<b>Figura 7</b> Ejemplo de aplicación de DCT en una imagen.....	20
<b>Figura 8</b> Selección en zig-zag de los d coeficientes.....	20
<b>Figura 9</b> Matriz de pesos usando método Simple-minded .....	22
<b>Figura 10</b> Matriz de alternativas de métodos de reconocimiento facial.....	25
<b>Figura 11</b> Enfoque PCA para reconocimiento facial .....	27
<b>Figura 12</b> Representación de la imagen .....	29
<b>Figura 13</b> Arquitectura de sistema de reconocimiento facial .....	32
<b>Figura 14</b> Preparación de la imagen.....	33
<b>Figura 15</b> Creación vector imagen .....	34
<b>Figura 16</b> Obtención de eigenvectors.....	35
<b>Figura 17</b> Obtención de eigenvalues .....	35
<b>Figura 18</b> Pantalla de inicio RF.....	36
<b>Figura 19</b> Entrenamiento.....	37
<b>Figura 20</b> Registro en BD .....	37
<b>Figura 21</b> Reconocimiento de un rostro nuevo .....	38
<b>Figura 22</b> Metodología SCRUM .....	42
<b>Figura 23</b> Panel de Sprint.....	43
<b>Figura 24</b> Modelo físico de la base de datos AUTOMATZER .....	62
<b>Figura 25</b> Definición de Actores StarUML.....	63
<b>Figura 26</b> Tareas por actor .....	63
<b>Figura 27</b> Casos de Uso General StarUML.....	64
<b>Figura 28</b> Diagrama de Secuencia Autentificación Usuario (Pin) .....	71
<b>Figura 29</b> Diagrama de Secuencia Autentificación Usuario (Reconocimiento Facial) .....	71
<b>Figura 30</b> Diagrama de Secuencia Realizar Solicitud.....	72
<b>Figura 31</b> Diagrama de Secuencia Seguimiento Solicitud .....	72
<b>Figura 32</b> Diagrama de Secuencia Histórico Solicitud .....	73
<b>Figura 33</b> Diagrama de Secuencia Responder Solicitud .....	73
<b>Figura 34</b> Diagrama de Secuencia Escalar Solicitud .....	74
<b>Figura 35</b> Diagrama de Secuencia Histórico Solicitud Pendiente.....	74
<b>Figura 36</b> Diagrama de Navegación Usuario Administrador.....	75
<b>Figura 37</b> Diagrama de Navegación Usuario Representante .....	75
<b>Figura 38</b> Diagrama de Navegación Usuario Soporte.....	76
<b>Figura 39</b> Arquitectura AUTOMATZER.....	76
<b>Figura 40</b> Explicación Objetivo del Proyecto .....	79
<b>Figura 41</b> Fotografía de entrenamiento al sistema .....	79
<b>Figura 42</b> Fotografía reconocimiento facial .....	80
<b>Figura 43</b> Realización de la Encuesta .....	80

<i>Figura 44</i> Cuadro estadístico pregunta 1.....	81
<i>Figura 45</i> Cuadro estadístico pregunta 2.....	82
<i>Figura 46</i> Cuadro estadístico pregunta 3.....	82
<i>Figura 47</i> Cuadro estadístico pregunta 4.....	83
<i>Figura 48</i> Cuadro estadístico pregunta 5.....	84
<i>Figura 49</i> Cuadro estadístico pregunta 6.....	85
<i>Figura 50</i> Cuadro estadístico pregunta 7.....	86
<i>Figura 51</i> Cuadro estadístico pregunta 8.....	87
<i>Figura 52</i> Cuadro estadístico pregunta 9.....	88
<i>Figura 53</i> Cuadro estadístico pregunta 10.....	89
<i>Figura 54</i> Rostros de prueba 1 .....	90
<i>Figura 55</i> Rostros de prueba 2 .....	92
<i>Figura 56</i> Cuadro estadístico de resultados segunda prueba.....	93

## **RESUMEN**

El presente trabajo de titulación tiene como objetivo la implementación de reconocimiento facial como mecanismo de seguridad para el sistema de soporte en línea destinado a los representantes de Pfizer. Para su desarrollo se inició con una investigación de diferentes métodos de reconocimiento facial y se seleccionó la más adecuada para el sistema. El método Eigenfaces, también conocido como PCA, fue implementado en JAVA en un sistema independiente solo de reconocimiento facial. Al mismo tiempo se inició el desarrollo del sistema de soporte en línea para representantes de Pfizer, utilizando la metodología de desarrollo ágil SCRUM. Por medio de reuniones con los involucrados, se definieron los requerimientos y se elaboró un prototipo del sistema que fue aprobado por un miembro de Pfizer. Las funciones del sistema consisten en recibir solicitudes de soporte por parte de un representante, procesar y responder esas solicitudes por parte de un miembro de soporte de la empresa. Posteriormente, el sistema de reconocimiento facial fue sometido a una prueba con estudiantes de la ESPE. El fin de esta prueba era encontrar un límite de comparación adecuado para utilizarlo dentro del algoritmo al intentar reconocer un rostro. Además, dentro de las pruebas se detectó una incidencia con las fotos de los rostros, estás no eran totalmente adecuadas para el reconocimiento. Con esto en cuenta, se realizaron los cambios necesarios al sistema para que la foto resultante se únicamente del rostro de la persona y se lo integró con el sistema de soporte, el cual fue aprobado por el Pfizer.

### **Palabras Clave:**

- **RECONOCIMIENTO FACIAL**
- **EIGENFACES**
- **PCA**
- **SISTEMA DE SOPORTE**

## **ABSTRACT**

The objective of this certification work is to implement facial recognition as a security mechanism for the online support system for Pfizer representatives. Its development began with an investigation of different methods and techniques of facial recognition and the most appropriate for the system was selected. The Eigenfaces method, also known as PCA, was implemented in JAVA in an independent system only for facial recognition. At the same time, the development of the online support system for Pfizer representatives began, using the agile development methodology SCRUM. Through meetings with those involved, the requirement were defined and a prototype of the system was developed and approved by a Pfizer member. The functions of the system consist of receiving requests for support from a representative, processing and responding to those requests by a support member of the company. Subsequently, the facial recognition system was tested with students of the University of the Armed Forces - ESPE. The purpose of this test was to find a suitable comparison limit to use within the algorithm when trying to recognize a face. In addition, within the tests an incidence was detected with the photos of the faces, these are not very suitable for recognition. With this in mind, the necessary changes for the system were made so that the resulting image was just the face of the person and this system was integrated with the support system, which was approved by the Pfizer.

**Keywords:**

- **FACIAL RECOGNITION**
- **EIGENFACES**
- **PCA**
- **SUPPORT SYSTEM**

## CAPÍTULO I

### INTRODUCCIÓN

En este capítulo se expondrán las ideas generales que motivan la realización de este trabajo de investigación, partiendo desde los antecedentes que permiten explicar el cómo se genera el problema. En el planteamiento del problema y justificación se explicará de forma específica el problema que se quiere resolver, las razones para hacerlo y lo que puede suceder en caso de no hacerlo. El problema planteado da lugar a la estructuración de los objetivos tanto específicos como general que se van a alcanzar con el desarrollo de este proyecto.

A continuación, el alcance será definido en dos partes que expresan lo que se desarrollará como parte de la solución al problema encontrado. Estas partes describirán la funcionalidad del sistema que se obtendrá como producto final. Como último punto de este capítulo, se formulará la hipótesis que se desea probar con el producto resultante de la realización de este trabajo de grado.

#### 1.1. Antecedentes

Pfizer es una importante empresa farmacéutica y es la compañía líder en investigación biomédica, cuya misión es descubrir y desarrollar medicinas innovadoras para ponerlas a disposición de la sociedad. Colabora, mediante sus representantes, en numerosos programas y actividades nacionales e internacionales, para mejorar la condición de vida de la población. Algunas de las actividades realizadas por los representantes de la empresa consisten en realizar visitas a médicos con el fin de promover la salud proporcionando sus productos farmacéuticos.

Los procesos que realizan los representantes de Pfizer generan un gran flujo de información sensible para la empresa que requiere de seguridad. Un ejemplo de esto son las visitas constantes a médicos por parte de los representantes para ofrecer sus productos, ya que se maneja información confidencial que debe ser protegida. Además, estas visitas deben ser registradas en la base de datos de la empresa y en este proceso suelen ocurrir errores que necesitan ser solucionados lo más pronto posible.

Por otro lado, para las empresas farmacéuticas el soporte a los representantes supone un arduo y costoso trabajo, ya que el proceso desde que ingresa la ayuda hasta que se la resuelve consta de varios pasos. Esto implica la manipulación directa de los datos de los médicos para el registro de las diferentes visitas, de manera que podrían ser alterados por personas no autorizadas. Hoy en día se contratan pasantes o empleados dedicados a responder a estas peticiones de ayuda, lo que representa un gasto adicional para este tipo de empresas. Por tal

motivo, es necesario establecer un proceso que permita mantener la integridad de la información y que reduzca tiempo y dinero en la solicitud de estas ayudas.

### **1.2. Planteamiento del problema**

Actualmente, Pfizer maneja dos tipos de canales. El primero es mediante llamada telefónica y el segundo es por correo electrónico. Debido a la demanda de solicitudes de ayuda en estos canales por parte de los representantes de Pfizer, la empresa tiene la necesidad de implementar un sistema de soporte para facilitar la atención.

La gestión que debe realizar el Área de Soporte SFA (Sales Force Automation) de los representantes lleva demasiado tiempo. Además, la información que éstos manejan es de vital importancia para el negocio. Por lo tanto, se pretende implementar un aplicativo que automatice la asistencia en casos como, incidentes con vistas médicas, problemas con otras aplicaciones utilizadas por los representantes, desbloqueo de contraseñas, entre otros, y reducir el tiempo de respuesta de estas peticiones.

Tomando en cuenta que los representantes manejan información delicada como datos personales y profesionales de los médicos, se necesita un medio de seguridad adecuado. Existen métodos comúnmente utilizados como son: credenciales de acceso, número de identificación personal (NIP), tarjeta de identificación por radiofrecuencia (RFID), token, etc., que requieren de recursos que son susceptibles a pérdidas, ocasionando problemas tanto económicos como de seguridad. (Saeed & Nagashima, 2012). Debido a que la seguridad es un factor clave en un sistema como el propuesto, es indispensable la implementación de un mecanismo que provea dicha característica.

### **1.3. Justificación**

El flujo de información que maneja Pfizer al brindar soporte a sus representantes requiere de una protección adecuada contra accesos no deseados. La seguridad de esta información debe ser controlada por algún mecanismo de acceso aceptable. En este caso, el reconocimiento facial dará un nivel de confiabilidad a un sistema de automatización de procesos que mejorará el tiempo de respuesta para las peticiones de los representantes de Pfizer y permitirá reducir gastos destinados al pago de pasantes que cubren las peticiones registradas.

Actualmente existen varias alternativas que podrían solventar el problema presentado, pero de forma individual a cada una de las soluciones propuestas. Es decir, se tendría un proyecto para la automatización de los procesos realizados por los representantes de Pfizer y

otro separado para optimizar la seguridad del sistema con reconocimiento facial. Adicionalmente, para este último fin es necesario realizar una investigación orientada a encontrar la técnica adecuada. El costo que involucraría la adquisición e implementación de estos dos proyectos por separado es sumamente alto y aumentaría al integrarlos en un solo sistema. Por tal motivo, este proyecto busca desarrollar un sistema que unifique estas dos soluciones en una para brindar una solución integral a bajo costo, pero de buena calidad que reduzca los tiempos de respuesta de peticiones y brinde la seguridad deseada.

En caso de no utilizar un sistema de automatización de procesos con un método de seguridad como el propuesto, Pfizer continuaría con los mismos gastos innecesarios en los que incurre para solventar las peticiones. Además, el proceso tal y como se lo viene realizando no garantiza que la información está siendo procesada de forma adecuada y con los niveles de seguridad necesarios. Por todo lo anterior expuesto, es necesario el desarrollo de este sistema, ya que será de utilidad para Pfizer en la realización de su actividad.

## **1.4. Objetivos**

### **1.4.1. Objetivo General**

Implementar reconocimiento facial como mecanismo de seguridad para el sistema de soporte en línea destinado a los representantes de Pfizer.

### **1.4.2. Objetivos Específicos**

- Investigar y analizar técnicas de reconocimiento facial para documentar su funcionamiento y seleccionar el adecuado.
- Implementar un algoritmo de reconocimiento facial, como medio de seguridad del sistema.
- Diseñar un sistema de soporte en línea para facilitar la ayuda y mejorar los tiempos de respuesta y la calidad del servicio prestado por el Área de Soporte de Pfizer.
- Evaluar el sistema tanto en seguridad como en funcionalidad para cumplir con los requerimientos de la empresa.

## **1.5. Alcance**

En primer lugar, se desarrollará un sistema que permita facilitar los procesos realizados por los representantes de Pfizer. Esto incluye:

- Reportar incidentes con alguna visita médica

- Reportar problemas con las aplicaciones MI<sup>1</sup> - ABM<sup>2</sup>
- Procesar los incidentes registrados en el orden en que se originan
- Dar solución a los incidentes reportados
- Escalar los casos complejos al equipo de soporte global (Colombia – India)
- Retornar una respuesta a la petición de ayuda

Esta parte del proyecto se llevará a cabo con la finalidad de automatizar el sistema y reducir los tiempos de respuesta antes las solicitudes de los representantes. La programación del sistema será desarrollada siguiendo los requerimientos de los involucrados para lograr un alto nivel de satisfacción.

Posteriormente, se implementará un mecanismo de seguridad basado en reconocimiento facial para el sistema antes descrito. Puntualmente, se analizarán distintas técnicas y algoritmos de reconocimiento de rostros y se seleccionará el apropiado para la implementación en el sistema, descartando aquellos que no vayan con la orientación del proyecto. Para este fin, se inicia con la búsqueda de información concerniente a los algoritmos antes mencionados. De los resultados obtenidos con esta búsqueda, se deberá documentar el funcionamiento de las técnicas utilizadas para proceder a seleccionar la mejor alternativa. Como punto final de esta etapa se utilizará la alternativa seleccionada para implementarla como método de seguridad en el sistema propuesto como solución al problema detallado en el [apartado 1.2](#).

## **1.6. Hipótesis**

H0: La utilización de un sistema de soporte en línea con reconocimiento facial como mecanismo de seguridad incrementará la productividad de las actividades realizadas por representantes de Pfizer y proporcionará un nivel de seguridad óptimo para la información involucrada.

## **1.7. Estructura del Trabajo de Investigación**

El presente trabajo de investigación estará organizado de la siguiente manera: i) En el capítulo II se darán a conocer las técnicas de reconocimiento facial, se analizarán los algoritmos principales de cada una de las técnicas, para finalmente seleccionar el algoritmo óptimo para su aplicación. ii) Una vez seleccionado el algoritmo, en el capítulo III se realizará la implementación del reconocimiento facial y se evaluará su funcionamiento para utilizarlo en el

<sup>1</sup> Herramienta perteneciente a CISCO para dar soporte al equipo de ventas de la empresa Pfizer.

<sup>2</sup> Herramienta que maneja la base de datos donde se encuentran los establecimientos y médicos a los cuales se ofrecen los productos.

sistema como medio de seguridad. iii) En el Capítulo IV se desarrollará el sistema de soporte con los requisitos receptados de Pfizer, y finalmente, en iv) el capítulo V se realizarán pruebas para ponerlo en producción.

## CAPÍTULO II

### TRABAJOS RELACIONADOS Y ANÁLISIS DE TÉCNICAS Y ALGORITMOS DE RECONOCIMIENTO FACIAL

Para iniciar con el análisis de las técnicas y algoritmos de reconocimiento facial es importante tener una definición clara del tema. (Parmar & Mehta, 2013) definen el reconocimiento facial como un sistema biométrico utilizado para identificar o verificar a una persona desde una imagen digital. Esto quiere decir que se utiliza la biometría para detectar automáticamente un rostro en una imagen. La biometría es la ciencia encargada de establecer la identidad de un individuo basado en atributos físicos, químicos o de comportamiento de la persona (Jain, Flynn, & Ross, 2008).

En este capítulo se darán a conocer técnicas de reconocimiento facial utilizadas a través de los años. Se expondrán sus características y se hará hincapié en las más utilizadas. Posteriormente, se seleccionarán las más relevantes para su evaluación dentro de un mismo escenario y se determinará la técnica o algoritmo más adecuado para el sistema propuesto.

#### **2.1. Técnicas de Reconocimiento Facial**

Según (Zhao, 2003), el reconocimiento facial es una de las aplicaciones de análisis y comprensión de imágenes más exitoso y ha recibido una atención significativa en diferentes conferencias y congresos científicos. Esto significa que, a lo largo de los años, el desarrollo de técnicas y algoritmos de reconocimiento facial ha evolucionado. A continuación, se presenta un resumen de la historia del reconocimiento facial, haciendo mención sobre algunos acontecimientos en los que esta tecnología fue utilizada.

**Tabla 1**  
*Historia del reconocimiento facial*

<b>(1960s)</b>	El padre del reconocimiento facial, Woodrow Wilson Bledsoe,
<b>Mediciones manuales por Bledsoe</b>	desarrolló un sistema que podía clasificar fotos de rostros manualmente utilizando una tablet RAND <sup>3</sup> . Este sistema se usaba para registrar manualmente las coordenadas de varias características del rostro en una base de datos. Cuando una nueva

Continúa 

---

<sup>3</sup> Dispositivos que puede ser utilizado para ingresar coordenadas horizontales y verticales en una cuadrícula por medio de un lápiz óptico que emite pulsos electromagnéticos.

	imagen ingresa al sistema, este retorna la imagen de la base de datos que más se parece al individuo de la nueva imagen.
<b>(1970s)</b> <b>Mayor precisión con 21 marcadores faciales</b>	Goldstein, Harmon, and Lesk lograron añadir una mayor precisión a un sistema manual de reconocimiento facial. Utilizaron 21 marcadores subjetivos específicos como ancho de labios y color del cabello para identificar rostros automáticamente.
<b>(Finales de los 80s)</b> <b>Inicios de los 90s)</b> <b>Eigenfaces</b>	En 1988, Sirovich y Kirby aplicaron álgebra lineal al reconocimiento facial. Esto se conoció como el enfoque Eigenface y comenzó como una búsqueda de una representación de baja dimensión de imágenes de rostros, que demostró que el análisis de características de una colección de imágenes puede formar un conjunto de características básicas. También demostraron que se requieren de menos de cien valores para codificar con precisión una imagen de rostro normalizada.  En 1991, utilizaron el enfoque Eigenface para descubrir cómo detectar rostros en imágenes. Este descubrimiento lideró las primeras instancias del reconocimiento facial automático.
<b>(1993-2000s)</b> <b>Programa FERET</b>	DARPA <sup>4</sup> y el inicio del programa FERET <sup>5</sup> en los años 90 para promover al reconocimiento facial en el mercado. El proyecto consistía en crear una base de datos de imágenes de rostros, que fue actualizada en 2003 para incluir versiones de imágenes de alta resolución con colores de 24 bits. El propósito era que con una base de datos extensa de imágenes de prueba para reconocimiento facial se genere innovación, que resultaría en la potencialización de esta tecnología.
<b>(2000s)</b> <b>Face Recognition Vendor Tests (FRVT)</b>	Las Pruebas de Proveedores de Reconocimiento Facial (FRVT) se desarrollaron en base a FERET para proveer evaluaciones gubernamentales independientes de sistemas de reconocimiento facial que estaban en el mercado. Estas pruebas se diseñaron para

Continúa 

<sup>4</sup> Defense Advanced Research Projects Agency

<sup>5</sup> National Institute of Standards and Technology program

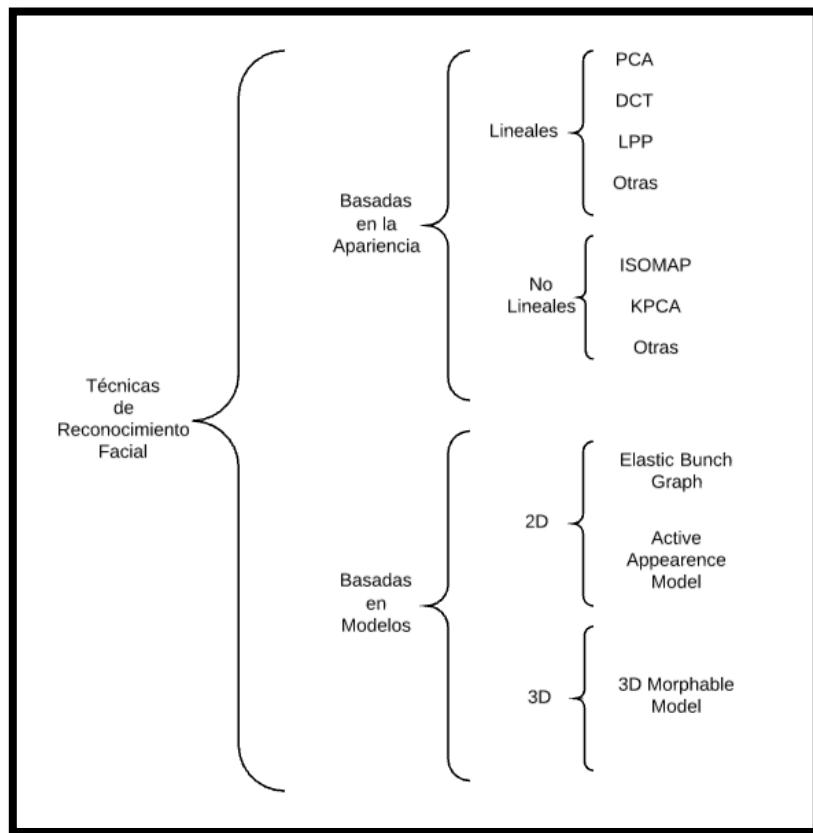
	brindar la información necesaria para determinar la mejor forma de implementar la tecnología de reconocimiento facial.
<b>(2002) Super Bowl XXXV</b>	Esta tecnología fue probada a gran escala en el Super Bowl del 2002. A pesar de que se detectaron algunos pequeños criminales, la prueba se consideró un fracaso. Una de las grandes limitaciones del reconocimiento facial en ese tiempo fue que no funcionaba con grandes multitudes.
<b>(2010-Presente) Redes Sociales</b>	En 2010 Facebook inició la implementación de reconocimiento facial que ayudaba a identificar personas cuyos rostros estaban en fotos que sus usuarios publicaban. Al no haber tenido un impacto negativo en el uso o popularidad del sitio, más de 350 millones de fotos son publicadas y etiquetadas cada día utilizando reconocimiento facial.
<b>(2011) Primera gran instalación de reconocimiento facial en un aeropuerto</b>	Un programa piloto de reconocimiento facial de FaceFirst fue autorizado por el gobierno de Panamá para terminar con la actividad ilícita en el aeropuerto de Tocumen. Esta implementación dio como resultado la captura de varios sospechosos de la Interpol.
<b>(2011) Identificación de Osama Bin Laden</b>	El reconocimiento facial se usó para ayudar a confirmar la identidad de Osama bin Laden después de que fuera asesinado en una redada en los EE. UU.
<b>(2017) Lista de seguimiento (watchlist) como servicio</b>	Watchlist como servicio es una plataforma de reconocimiento facial diseñada para ayudar a prevenir el hurto y el crimen violento. Esta lista incluye una base de datos de criminales conocidos y trabaja a la par con la plataforma de vigilancia biométrica de FaceFirst que utiliza tecnología de emparejamiento por características para alertar sobre amenazas en tiempo real.

Fuente: (Davis West, 2017)

El reconocimiento facial es una tecnología en desarrollo constante que, como se resume en la Tabla 1, desde su inicio ha sido utilizada en varios eventos de relevancia alrededor del mundo como la identificación de Osama Bin Laden. El éxito o fracaso de su uso en los eventos

mencionados se debió básicamente a limitaciones en los avances tecnológicos de la época. Sin embargo, otro factor importante a tomar en cuenta es la técnica utilizada en la implementación.

De acuerdo con (Lu, 2003), en las últimas décadas se han desarrollado y modificado diversos algoritmos de reconocimiento facial, los cuales se categorizan en dos técnicas: basadas en la apariencia y basadas en modelos, como se observa en la Figura 1. En las técnicas basadas en apariencia se utilizan varias estadísticas obtenidas de los píxeles de la imagen del rostro (Li & Jain, 2009), es decir, representan un objeto en función de diferentes vistas del mismo. En cambio, el reconocimiento facial basado en modelos está dirigido a construir un modelo del rostro humano que es capaz de capturar las variaciones del mismo (Lu, 2003).



**Figura 1** Técnicas de reconocimiento facial

Fuente: (Lu, 2003)

Tomando en cuenta los aspectos positivos que expone (Hernández, 2010) y que se ha realizado un compendio de éstas en la Tabla 2, el presente proyecto hará uso de las técnicas basadas en apariencia, de las cuales se profundizará a detalle más adelante.

**Tabla 2**

*Ventajas e Inconvenientes de las técnicas basadas en apariencia*

<b>Ventajas</b>	<b>Inconvenientes</b>
Se requiere un menor tamaño de las imágenes	Los cambios en la orientación o expresión de la cara afectan la exactitud.
Son de menor complejidad	Cambios en la iluminación
No se requiere de conocimiento previo de las imágenes	
Las técnicas basadas en apariencias presentan resultados en un menor tiempo.	
Compara dos caras proyectando las imágenes y midiendo la distancia entre ellas.	
Trabaja bien con alta dimensión.	
En el método basado en el componente principal Análisis (PCA reduce la dimensión de los datos)	

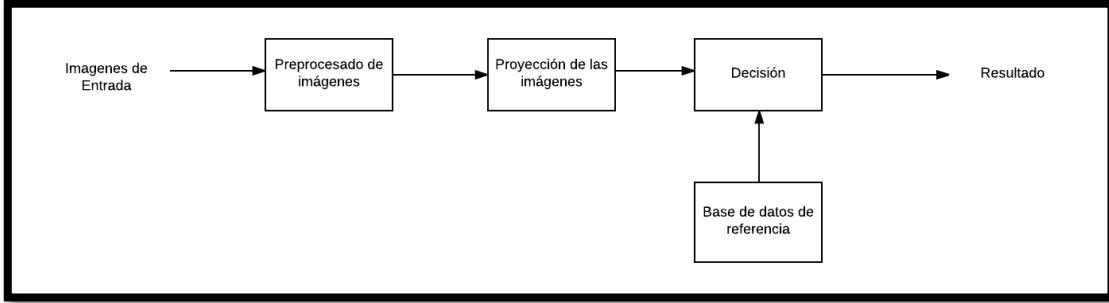
Fuente: (Hernández, 2010)

### Técnicas de Basadas en la Apariencia

Las técnicas basadas en la apariencia se centran en métodos estadísticos y el aprendizaje automático para encontrar las características relevantes de las imágenes faciales. Algunos métodos basados en apariencia funcionan en una red probabilística (Eleyan & Demirel, 2007). Una imagen o vector de característica es una variable aleatoria con alguna probabilidad de pertenecer a un rostro o no. Estos métodos también pueden ser usados en la extracción de características para el reconocimiento facial.

El proceso realizado por las técnicas basadas en apariencia para realizar el reconocimiento facial inicia con la imagen que se desea analizar. Esta imagen pasa por un pre

proceso en el que se encuentran sus características relevantes para proyectarlas a la etapa de toma de decisión. Para la decisión se hace uso de una base de datos de referencia en el que se encuentran las características de las imágenes que se utilizaran para la comparación. Por último, se retorna el resultado de si el reconocimiento fue exitoso o no. En la Figura 2 se puede observar este proceso de forma gráfica.



**Figura 2** Diagrama de bloques

Fuente: (Hernández, 2010)

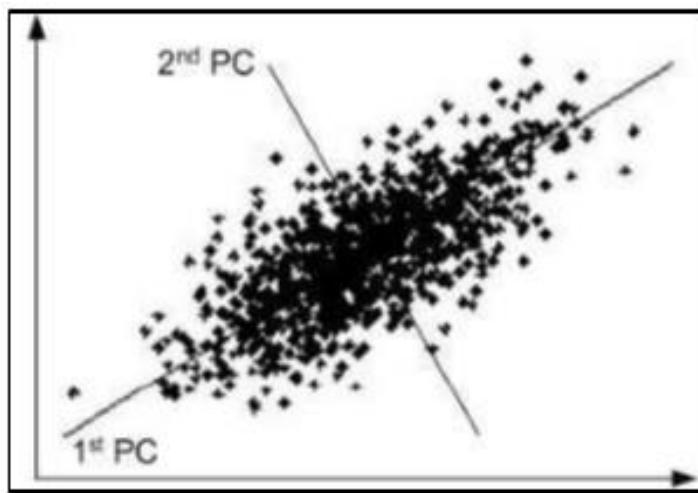
Las técnicas basadas en la apariencia se clasifican en lineales y no lineales. De las técnicas lineales se profundizará en PCA, LDA, DCT, LPP, y dentro de las técnicas no lineales se analizará ISOMAP. Se seleccionaron estas técnicas debido a que son las más utilizadas dentro del reconocimiento facial.

### 2.1.1. Análisis de Componentes Principales (PCA)

El análisis de componentes principales, también conocido como eigenfaces (Turk & Pentland, 1991), es una forma de identificar patrones en los datos y expresarlos de tal manera que resalten sus similitudes y diferencias. La PCA es un método muy poderoso al momento de analizar datos, ya que los patrones en los datos pueden ser difíciles de encontrar cuando se tiene mayor información (Eleyan & Demirel, 2007).

Una ventaja de PCA, según (Chiang, Hsieh, & Dhillon, 2016), es que una vez que ha encontrado estos patrones en los datos, los comprime, es decir reduce el número de dimensiones sin mucha pérdida de información. De este modo, se reduce el tamaño de la base de datos para el reconocimiento de una imagen de prueba, ya que las imágenes se almacenan como vectores de características en la base de datos para así reducir la dimensionalidad de un conjunto de datos.

El propósito de este método consiste en representar una imagen en términos de un sistema de coordenadas óptimo reduciendo el número final de componentes que tendrá la imagen. Un ejemplo de PCA se puede observar en la Figura 3.



**Figura 3** Componentes principales de conjunto de puntos bidimensional  
**Fuente:** (Lu, 2003)

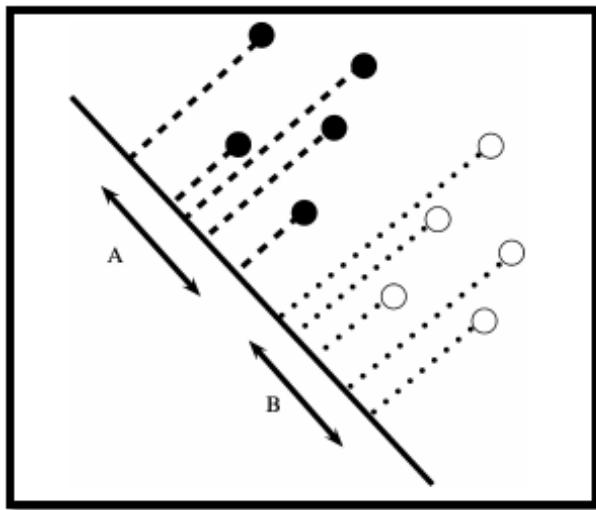
Los enfoques basados en PCA comúnmente se dividen en dos fases: entrenamiento y clasificación. De acuerdo con (Eleyan & Demirel, 2007), en la fase de entrenamiento se establece un eigenspace<sup>6</sup> a partir de las muestras de entrenamiento y las imágenes faciales de entrenamiento son mapeadas al eigenspace para su clasificación. En la fase de clasificación, un rostro de entrada es proyectado al mismo eigenspace y clasificado por un clasificador apropiado.

### 2.1.2. Análisis de Discriminante Lineal (LDA)

El Análisis de Discriminante Lineal (LDA) según (Xanthopoulos, Pardalos, & Trafalis, 2013), es un método de análisis de datos fundamental originalmente propuesto por R. Fisher para discriminar entre diferentes tipos de flores. La intuición detrás del método es determinar un subespacio de dimensión inferior, en comparación con la dimensión de muestra de datos original, en la que los puntos de datos del problema original son "separables" como se observa en la Figura 4.

---

<sup>6</sup> Espacio generado por un eigenvector y su correspondiente eigenvalue - es decir, el espacio de todos los vectores que pueden ser descritos como combinaciones lineales de dichos eigenvectors.



**Figura 4** Separabilidad de las muestras

proyectadas en dos dimensiones

Fuente: (Xanthopoulos, Pardalos, & Trafalis, 2013)

Una de las ventajas de LDA es que la solución puede obtenerse resolviendo un sistema de auto valor generalizado. Esto permite un procesamiento rápido y masivo de muestras de datos. Además, LDA puede extenderse a LDA a través del truco del kernel. El algoritmo original se propuso para problemas de clase binarios, pero también se han propuesto generalizaciones de clases múltiples

### 2.1.3. Transformada de Coseno Discreta (DCT)

La transformada de coseno discreta (DCT) según (Fracastoro, Fosson, & Magli, 2017), es una transformación significativa ampliamente utilizada en el procesamiento digital de imágenes. Las cantidades desmesuradas de DCT se enfocan en la sección de baja frecuencia; en lo sucesivo, se reconoce que tiene una gran vitalidad de firmeza y posesiones, por lo que la matriz de transformación es fija e independiente de la imagen.

Una de las principales características de la DCT que lo hace atractivo para el reconocimiento facial es el cálculo eficiente, debido que tiene relación con la transformada de Fourier discreta, es decir transforma una señal o imagen del dominio espacial al dominio de frecuencia. Además, a diferencia de PCA, la DCT no necesita de un entrenamiento previo con otras imágenes.

### 2.1.4. Proyecciones Locales de Preservación (LPP)

Proyecciones locales de preservación (LPP) según (He & Niyogi, 2004) son mapas proyectivos lineales que surgen al resolver un problema variacional que preserva de manera óptima la estructura de vecindad del conjunto de datos. LPP debe verse como una alternativa al

Análisis de Componentes Principales (PCA), ya que, así como PCA requiere de un entrenamiento previo de imágenes.

### **2.1.5. Mapeo de Características Isométricas (ISOMAP)**

En los últimos años, el aprendizaje múltiple para la reducción de dimensionalidad no lineal (Zang, 2011) y la visualización de datos ha atraído mucha atención. El mapeo de características isométricas (Isomap) según (Tenenbaum, Silva, & Langford, 2000), es uno de los métodos más populares para el aprendizaje múltiple. Las ventajas de Isomap incluyen la optimalidad global, las garantías de convergencia asintótica y la aplicabilidad a una amplia clase de variedades no lineales. Debido a su excelente rendimiento, es ampliamente utilizado en muchos campos, como reconocimiento de patrones (Yazdian, Tie, & Venetsanopoulos, 2014), procesamiento de señales (Gepshtein & Keller, 2015), procesamiento de imágenes (Verma, Khurd, & Davatzikos, 2007), meteorología (Fallah & Sodoudi, 2015) y neurociencia.

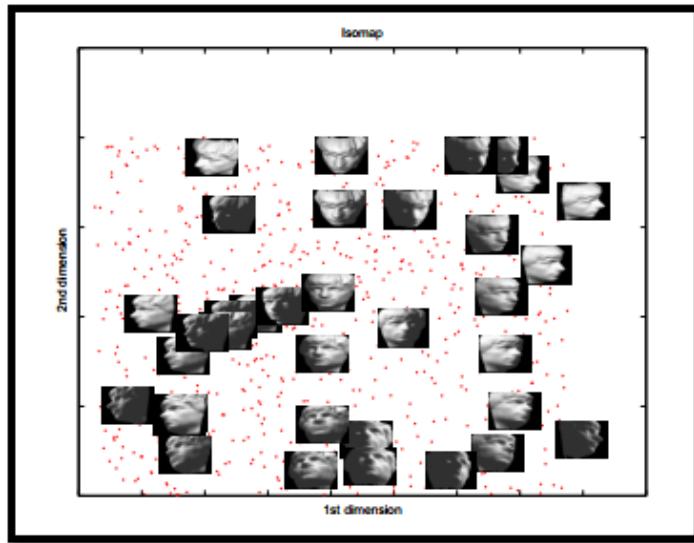
Isomap se ve como una extensión de la escala multidimensional clásica (CMDS). Según (Cox & Cox, 2001), se basa en CMDS y busca aproximar las distancias geodésicas entre todos los pares de puntos de datos por las distancias en un espacio euclíadiano de baja dimensión. Las distancias geodésicas entre puntos vecinos se aproximan por la distancia del espacio de entrada. Para puntos lejanos, las distancias geodésicas se aproximan por caminos más cortos en un gráfico de vecindad. Las coordenadas de baja dimensión de los puntos de datos se descubren aplicando CMDS a la matriz de distancia geodésica

Según (Ghodsi, 2006) Isomap es una generalización no lineal de MDS clásico. La idea principal es realizar MDS<sup>7</sup>, no en el espacio de entrada, sino en el espacio geodésico de la variedad de datos no lineales. Las distancias geodésicas representan las trayectorias más cortas a lo largo de la superficie curva del colector, medidas como si la superficie fuese plana.

---

<sup>7</sup> Multidimensional scaling.- Es un enfoque clásico que asigna el espacio de alta dimensionalidad original a un espacio dimensionalidad inferior, pero lo hace con la intención de preservar las distancias pares.

Para un ejemplo de Isomap se observa la Figura 5.



**Figura 5** Ejemplo de Isomap  
Fuente: (Ghodsi, 2006)

El algoritmo de Isomap consta de tres pasos que se mencionan a continuación:

- Encontrar los vecinos de cada punto de datos en el espacio de datos de alta dimensión.
- Calcular las distancias geodésicas por parejas entre todos los puntos.
- Introducir los datos a través de MDS para conservar estas distancias.

## 2.2. Análisis de algoritmos de Reconocimiento Facial

En este apartado se hablará del proceso y fórmulas matemáticas empleadas en el desarrollo de los métodos de reconocimiento facial anteriormente expuestos. Se omitirá el método ISOMAP debido a que utiliza procedimientos no lineales para realizar el reconocimiento del rostro. Una vez expuestos el procedimiento empleado por cada método, se realizará la selección de la técnica que será implementada en el sistema de soporte propuesto.

Las técnicas y métodos analizados están estrechamente relacionados con los algoritmos de reconocimiento facial. Cada algoritmo utiliza una técnica para analizar y comparar una imagen con otra., Por ejemplo, el algoritmo de Eigenfaces utiliza la técnica de PCA y el algoritmo de Fisherfaces utiliza LDA.

### 2.2.1. PCA - Análisis de Componentes Principales (Eigenfaces)

(Hernández, 2010) afirma que para realizar el análisis de rostros con PCA se debe realizar una matriz de proyección utilizando el siguiente procedimiento:

## Creación de la matriz de proyección

Se entrena PCA con  $N$  imágenes ( $X_i$ ), las mismas que deben ser de diferentes personas con diferentes puntos de vista, es decir frontal, de perfil, con diferentes faciales, con gafas, etc.  $\underline{W}$  (Matriz de Proyección) se calcula como sigue:

1. Calcular la media del vector.

$$\underline{\mu} = \sum_{i=1}^N \underline{x}_i \quad (1)$$

2. Calcular la matriz de Covarianza.

$$\underline{\underline{S}}_T = \sum_{i=1}^N (\underline{x}_i - \underline{\mu}) \cdot (\underline{x}_i - \underline{\mu})^T \quad (2)$$

3. Calcular los auto vectores y los auto valores de  $\underline{\underline{S}}_T$  y generar  $\underline{W}$  para que sus columnas se conviertan en auto valores más significativos.

$$\underline{\underline{S}}_T \Rightarrow \underline{\underline{W}} = (\underline{e}_1 \ \underline{e}_2 \ \dots \ \underline{e}_M) \quad (3)$$

## Reconocimiento Facial

Para poder entrenar una nueva imagen de prueba se realiza los siguientes pasos.

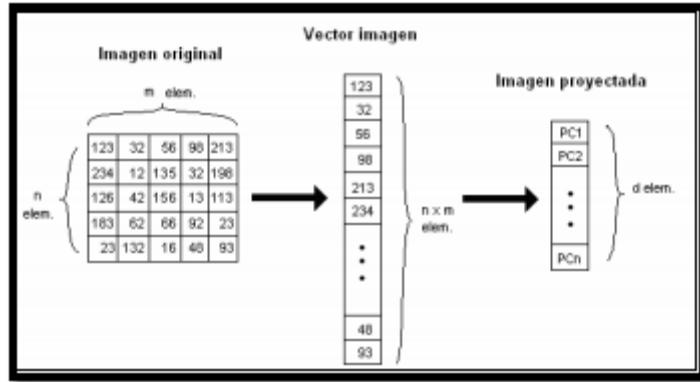
1. Primero, se redimensiona la imagen de prueba que se utilizó para la creación de la matriz de proyección. En el trabajo realizado por (Hernández, 2010) se utilizaron imágenes de tamaño 32x32 pixeles.
2. Se normaliza en media 0 y norma 1.
3. Se quita la media de la clase PCA por filas.

Luego de haber realizado estos tres pasos, la imagen está lista para proyectarse al nuevo subespacio. En este procedimiento, cada entrada al sistema  $x$  se proyecta al subespacio “rostro” por medio del método PCA, tal como se muestra en la siguiente ecuación.

$$\underline{y} = \underline{\underline{W}}^T \cdot \underline{x} \quad (4)$$

Donde  $y$  es la imagen proyectada en el nuevo subespacio de baja dimensionalidad,  $\underline{W}$  es la matriz de proyección PCA y  $x$  es la imagen de entrada.

Luego de realizar esta proyección, el vector de entrada n-dimensional se reducirá a un vector d-dimensional del subespacio, tal como se puede observar en la Figura 6.



**Figura 6** Reducción dimensional al aplicar PCA

Fuente: (Hernández, 2010)

### 2.2.2. LDA - Análisis de Discriminante Lineal (Fisherfaces)

Para realizar el Análisis de Discriminante Lineal según (Xanthopoulos, Pardalos, & Trafalis, 2013), se realiza los siguientes pasos:

Sea  $x_1, \dots, x_p \in R_m$  un conjunto de muestras de datos  $p$  que pertenecen a dos conjuntos de clases diferentes,  $A$  y  $B$ . Para cada clase podemos definir la siguiente fórmula:

$$\bar{x}_A = \frac{1}{N_A} \sum_{x \in A} x, \quad \bar{x}_B = \frac{1}{N_B} \sum_{x \in B} x \quad (5)$$

Donde  $N_A, N_B$  son el número de muestras en  $A$  y  $B$ , respectivamente. Luego, para cada clase, podemos definir las matrices de dispersión semidefinida positivas descritas por las ecuaciones:

$$S_A = \sum_{x \in A} (x - \bar{x}_A)(x - \bar{x}_A)^T, \quad S_B = \sum_{x \in B} (x - \bar{x}_B)(x - \bar{x}_B)^T \quad (6)$$

Cada una de estas matrices expresa la variabilidad de la muestra en cada clase. Idealmente, se busca encontrar un hiperplano, definido por el vector  $\phi$ , para lo cual, si se proyectan las muestras de datos, su varianza sería mínima. Eso se puede expresar como:

$$\min_{\phi} (\phi^T S_A \phi + \phi^T S_B \phi) = \min_{\phi} \phi^T (S_A + S_B) \phi = \min_{\phi} \phi^T S \phi \quad (7)$$

Donde  $S = S_A + S_B$  por definición. Por otro lado, la matriz de dispersión entre las dos clases está dada por:

$$S_{AB} = (\bar{x}_A - \bar{x}_B)(\bar{x}_A - \bar{x}_B)^T. \quad (8)$$

De acuerdo con la intuición de Fisher, se desea encontrar un hiperplano para maximizar la distancia entre los medios entre las dos clases y al mismo tiempo para minimizar la varianza en cada clase. Matemáticamente, esto se puede describir mediante la maximización del criterio de Fisher:

$$\max_{\phi} \mathcal{J}(\phi) = \max_{\phi} \frac{\phi^T S_{AB} \phi}{\phi^T S \phi}. \quad (9)$$

Este problema de optimización puede tener infinitas soluciones con el mismo valor de función objetivo. Eso es para una solución  $\phi$  todos los vectores  $c \cdot \phi$  dan exactamente el mismo valor. Para tener un valor sin pérdida de generalidad, se reemplaza el denominador con una restricción de igualdad para elegir una sola solución. Entonces el problema se convierte en:

$$\max_{\phi} \phi^T S_{AB} \phi \quad (10)$$

$$\text{s.t. } \phi^T S \phi = 1 \quad (11)$$

El lagrangiano asociado con este problema es:

$$\mathcal{L}_{LDA}(x, \lambda) = \phi^T S_{AB} \phi - \lambda (\phi^T S \phi - 1) \quad (12)$$

Donde  $\lambda$  es el multiplicador de lagrange que está asociado con la restricción (11). Como  $S_{AB}$  es positivo semidefinido, el problema es convexo y el mínimo global estará en el punto para el cual

$$\frac{\partial \mathcal{L}_{LDA}(x, \lambda)}{\partial x} = 0 \Leftrightarrow S_{AB} \phi - \lambda S \phi = 0 \quad (13)$$

El óptimo  $\phi$  se puede obtener como el vector propio que corresponde al valor propio más pequeño del siguiente eigensistema generalizado:

$$S_{AB} \phi = \lambda S \phi \quad (14)$$

LDA es una extensión natural del caso anterior. Dadas  $n$  clases, necesitamos redefinir las matrices de dispersión: la matriz intraclase se convierte

$$S = S_1 + S_2 + \dots + S_n \quad (15)$$

mientras que la matriz de dispersión entre clases está dada por

$$S_{1,\dots,n} = \sum_{i=1}^n p_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T \quad (16)$$

Donde  $p_i$  es el número de muestras en la clase  $i$ -ésima,  $x_i$  es la media de cada clase, y  $\bar{x}$  es el vector promedio total calculado por

$$\bar{x} = \frac{1}{P} \sum_{i=1}^P p_i \bar{x}_i. \quad (17)$$

La transformación lineal que deseamos encontrar se puede obtener resolviendo el siguiente problema de autovalores generalizados:

$$S_{1,\dots,n} \phi = \lambda S \phi. \quad (18)$$

LDA se puede usar para identificar cuáles son las características más importantes junto con el nivel de significancia expresado por el coeficiente correspondiente del hiperplano de proyección. También LDA puede usarse para clasificar muestras desconocidas. Una vez que se da la transformación  $\phi$ , la clasificación se puede realizar en el espacio transformado en función de alguna medida de distancia  $d$ . La clase de un nuevo punto  $z$  está determinada por:

$$\text{class}(z) = \arg \min_n \{d(z\phi, \bar{x}_n \phi)\} \quad (19)$$

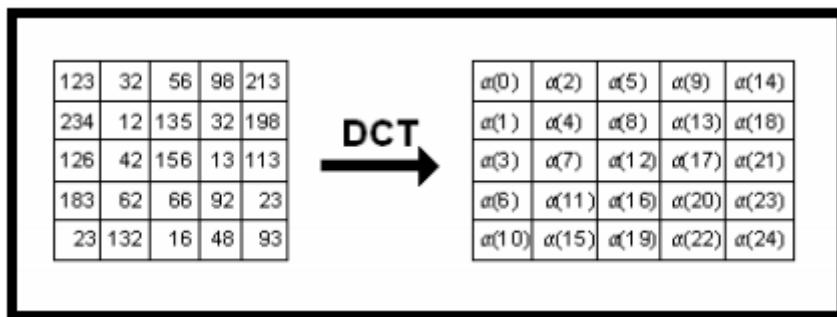
Donde  $x_n$  es el centroide de la  $n$ -ésima clase. Esto significa que primero proyectamos los centroides de todas las clases y los puntos desconocidos en el subespacio definido por  $\phi$  y asignamos los puntos a la clase más cercana con respecto a  $d$ .

### 2.2.3. DCT - Transformada de Coseno Discreta

#### Reconocimiento Facial

Para poder entrenar una nueva imagen de prueba se realiza los siguientes pasos.

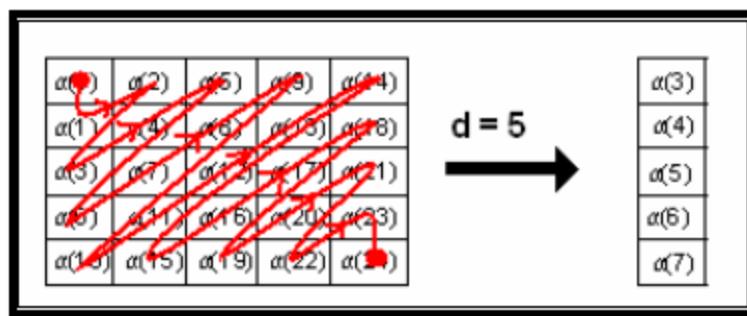
1. Tal como se había mencionado en el método PCA la imagen se escala a un tamaño de 32 x 32 pixeles.
2. Se calcula la DCT-II de la imagen escalada original como se puede apreciar en la Figura 7.



**Figura 7** Ejemplo de aplicación de DCT en una imagen

Fuente: (Hernández, 2010)

1. Los  $d$  coeficientes que se utilizarán para el reconocimiento se seleccionan en forma de zig-zag y se forma un vector columna, como se observa en la Figura 8. Para realizar esta selección, no se toma en cuenta  $a(0)$  el cual representa el componente continuo, y se pueden descartar los primeros coeficientes, para tener más control sobre los cambios de luminosidad, ya que son los que más se afectan por este tipo de modificaciones.



**Figura 8** Selección en zig-zag de los  $d$  coeficientes

Fuente: (Hernández, 2010)

1. La imagen de prueba se compara con la imagen de referencia y se toma una decisión. Para hacer esto, se usa de nuevo el criterio de distancia, en este caso la distancia Euclídea. De nuevo, el criterio de decisión es la distancia mínima.

## 2.2.4. LPP - Proyecciones Locales de Preservación

Para poder realizar el análisis, así como en el método de PCA según (Hernández, 2010) en LPP se debe realizar una matriz de proyección, para lo cual a continuación se dará a conocer su procedimiento.

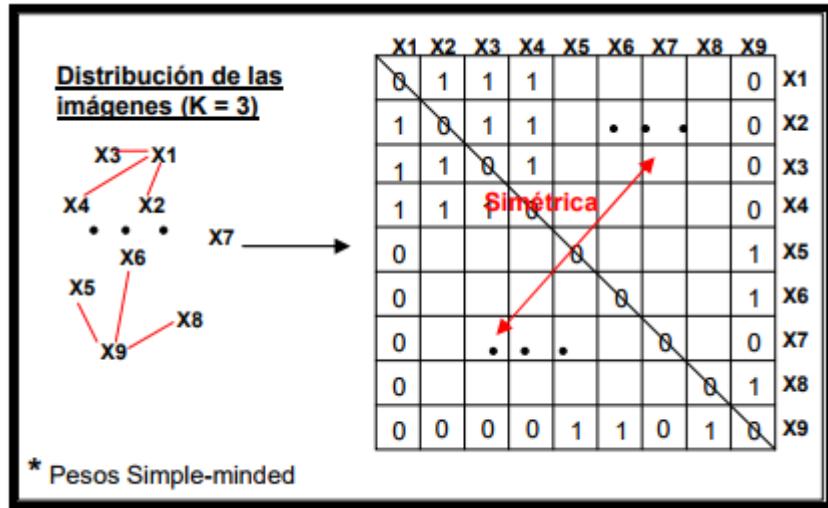
### Creación de la matriz de proyección

Para realizar la creación de la matriz de proyección, se sigue el mismo procedimiento que en el método de PCA, es decir el sistema se entrenará con  $N$  imágenes ( $X_i$ ) de diferentes personas, tomadas bajo diferentes condiciones. El procedimiento seguido para encontrar  $A$  (matriz de proyección) según (Hernández, 2010) consta de los siguientes pasos:

1. Las imágenes se procesan de la misma manera que para PCA en la etapa de reconocimiento (pasos 1,2,3 del [apartado 2.2.1](#)).
2. Las imágenes se proyectan usando la matriz de proyección PCA para evitar matrices singulares.
3. Una vez que mediante PCA se proyectan las imágenes, se construye un gráfico de adyacencia. Este gráfico tiene la forma en que las imágenes consideradas como vecinas poseen una conexión entre ellas. Según (Hernández, 2010) se pueden diferenciar 2 casos:
  - a.  $\varepsilon$ -vecinos. (El parámetro  $\varepsilon$  pertenece a  $R$ ) Los nodos  $i$  y  $j$  están conectados si  $\|x_i - x_j\|_2 < \varepsilon$ , donde la norma es la norma Euclídea usual en  $R_n$ .
  - b. K-nearest neighbors. (k pertenece a N). Los nodos  $i$  y  $j$  están conectados entre sí en el caso en que  $i$  sea de los  $k$  vecinos más cercanos a  $j$  o  $j$  sea de los  $k$  vecinos más cercanos de  $i$ .
4. Se deben seleccionar los pesos, existen 2 métodos para asignar estos pesos. La matriz que se crea,  $W$ , es una matriz simétrica de  $d \times d$  elementos que tienen un valor, dependiendo del método que se usa, si hay una conexión entre los elementos  $i$  y  $j$  en el gráfico de adyacencia o cero en caso que no haya conexión. Los métodos utilizados según (Hernández, 2010) son:
  - a. Heat Kernel. Necesita un parámetro  $t$  que el usuario puede generar en caso de que los nodos  $i$  y  $j$  están conectados, se puede asignar un elemento al valor:

$$W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}} \quad (20)$$

- b. Simple-minded. Este es el caso binario, es decir, si hay conexión entre los nodos  $i$  y  $j$ , se asigna 1 a ese elemento ( $W_{ij}$ ), o de lo contrario se asignará 0 como se puede apreciar en la Figura 9:



**Figura 9** Matriz de pesos usando método Simple-minded

Fuente: (Hernández, 2010)

5. Resolver la ecuación de auto valores. Se calculan los auto vectores y auto valores correspondientes a la solución de la ecuación general de auto vectores. Para esto, se usa una matriz diagonal (D), que indica la relevancia de los datos y, se forman a partir de las columnas de  $W$ , o por filas que son simétricas  $W$ , y de la matriz de Laplaciana (L). Una vez que se resuelve la ecuación, obtuvimos los auto vectores  $a_0, \dots, a_{l-1}$  ordenados de menor a mayor auto valor ( $\lambda_0 < \lambda_1 < \dots < \lambda_l$ ), dado que es posible observar una diferencia en PCA, se usan los auto vectores correspondientes a los auto valores más bajos.

$$D_{ii} = \sum_j W_{ji} \quad (21)$$

$$L = D - W \quad (22)$$

$$X \cdot L \cdot X^T \cdot a = \lambda \cdot X \cdot D \cdot X^T \cdot a \quad (23)$$

Finalmente se obtiene la siguiente fórmula.

$$x_i \rightarrow y_i = A^T \cdot x_i, \quad A = (a_0, a_1, \dots, a_{l-1}) \quad (24)$$

Donde  $y$  es un vector de  $l$  elementos de dimensión, y  $A$  es una matriz de  $n \times l$  elementos.

## Reconocimiento Facial

Para el entrenamiento de una nueva imagen de prueba se realiza los siguientes pasos.

1. Las imágenes se procesan de la misma manera que para PCA (pasos 1,2,3 del [apartado 2.2.1](#)).
2. Las imágenes son proyectadas. Para realizar este paso, se debe utilizar la matriz de proyección PCA y el LPP para que la imagen en el nuevo subespacio siga la siguiente expresión:

$$\underline{y} = \left( \underline{\underline{W}_{PCA}} \cdot \underline{\underline{A}_{LPP}} \right)^T \cdot \underline{x} \quad (25)$$

3. Se aplica un método de decisión. En el caso de este estudio, de la misma manera que en otros métodos, se aplicará el criterio de distancia mínima.

La importancia de LPP es que se define en todas partes en el espacio ambiental y no solo en los puntos de datos de entrenamiento. LPP puede llevarse a cabo en el espacio original o en el kernel de reproducción del espacio de Hilbert en el que se asignan los puntos de datos.

### 2.3. Selección de método de reconocimiento facial para la implementación en el sistema de soporte

Para seleccionar el método de reconocimiento facial adecuado para el sistema de soporte que se va a desarrollar como producto final de este proyecto, se tomaron en cuenta los siguientes criterios de decisión: i) complejidad matemática, ii) entrenamiento previo, iii) base de datos y iv) procesamiento y toma de decisión, que se encuentran detallados en la Figura 10. Las escalas de calificación de cada criterio son las siguientes:

**Tabla 3**  
*Escala de complejidad matemática*

CALIFICACIÓN	DETALLE
<b>1</b>	Muy alta complejidad
<b>2</b>	Alta complejidad
<b>3</b>	Complejidad media
<b>4</b>	Baja complejidad
<b>5</b>	Sin complejidad

**Tabla 4**  
*Escala de entrenamiento previo*

CALIFICACIÓN	DETALLE
<b>1</b>	Entrenamiento muy complejo
<b>2</b>	Entrenamiento complejo
<b>3</b>	Entrenamiento medio
<b>4</b>	Entrenamiento sencillo
<b>5</b>	No necesita entrenamiento

**Tabla 5**  
*Escala de base de datos*

CALIFICACIÓN	DETALLE
<b>1</b>	Cantidad de datos almacenados muy grande
<b>2</b>	Cantidad de datos almacenados grande
<b>3</b>	Cantidad de datos almacenados media
<b>4</b>	Cantidad de datos almacenados baja
<b>5</b>	Cantidad de datos almacenados muy baja

**Tabla 6***Escala de procesamiento y toma de decisión*

CALIFICACIÓN	DETALLE
<b>1</b>	Muy Ineficiente
<b>2</b>	Ineficiente
<b>3</b>	Medianamente eficiente
<b>4</b>	Eficiente
<b>5</b>	Muy eficiente

Se utilizaron estos criterios debido a que son los parámetros más importantes en el correcto funcionamiento de cada uno de las técnicas analizadas. Los métodos involucrados son los mismos que fueron analizados en el [apartado 2.2](#).

MÉTODOS	CRITERIOS DE DECISIÓN		COMPLEJIDAD MATEMÁTICA		ENTRENAMIENTO PREVIO		BASE DE DATOS		PROCESAMIENTO Y TOMA DE DECISIÓN		SUMA TOTAL
			CALIFICACIÓN	PENDERACIÓN	CALIFICACIÓN	PENDERACIÓN	CALIFICACIÓN	PENDERACIÓN	CALIFICACIÓN	PENDERACIÓN	
			0,4		0,1		0,3		0,2		
PCA	3	1,2	3	0,3	4	1,2	4	0,8	3,5		
LDA	2	0,8	3	0,3	3	0,9	4	0,8	2,8		
DCT	1	0,4	5	0,5	5	1,5	4	0,8	3,2		
LPP	2	0,8	3	0,3	4	1,2	3	0,6	2,9		

**Figura 10** Matriz de alternativas de métodos de reconocimiento facial

Cada criterio tiene asignada una calificación, una ponderación y un peso que sumado al de los demás da un total de 1. La calificación puede ser un número de 1 a 5 y fue dada por los desarrolladores de este proyecto, en base a la investigación realizada de cada técnica. La ponderación se obtiene con la multiplicación del peso de cada criterio por la calificación de cada técnica analizada en dicho criterio. Al final se obtuvo la suma total de las ponderaciones y se eligió la técnica con la suma más alta.

La complejidad matemática del método que se implementará debe ser acorde a los conocimientos adquiridos en la carrera, por lo cual, es el primer criterio analizado con un peso de 0.4. La calificación de 1 significa que la complejidad de la técnica es muy alta y 5 significa que el método no es complejo. Al método PCA se le otorgó la calificación más alta debido a

que las funciones matemáticas necesarias son parte del área de álgebra lineal impartida durante la carrera de Ingeniería en Sistemas e Informática.

El criterio de entrenamiento previo de imágenes, consiste en si el método necesita ser previamente entrenado para su correcto funcionamiento. El entrenamiento radica en proyectar las características de un nuevo rostro al conjunto de muestra para encontrar similitudes. En caso de no necesitar entrenamiento la calificación será de 5 y las demás calificaciones fueron dadas en base a la complejidad del proceso de entrenamiento necesario.

El criterio de base de datos, tiene relación con la cantidad de datos que se requiere almacenar, si esta es mayor, la calificación será 1, caso contrario aumentará progresivamente. El método DCT tiene la mejor calificación, ya que, al no necesitar un entrenamiento previo, requiere de menos información sobre las imágenes de rostros.

El criterio final es el procesamiento y toma de decisión, que tiene que ver con el tiempo que le toma al método en realizar el reconocimiento y retornar una respuesta. La calificación de 1 significa que el tiempo es menos eficiente, caso contrario, se le asignará una mejor calificación. Los métodos PCA, LDA y DCT tienen un grado de procesamiento similar por lo cual se les calificó con 4, siendo esta la mejor calificación de este criterio.

Finalmente, la suma más alta de las ponderaciones es para PCA, que es la técnica seleccionada. PCA corresponde al algoritmo Eigenfaces cuya implementación se analizará en el [Capítulo III](#).

En este capítulo se analizaron cinco diferentes técnicas de reconocimiento facial que utilizaban métodos de procesamientos lineales y no lineales. Debido a que se tiene un mayor dominio de los métodos lineales, se descartó la técnica ISOMAP que utiliza métodos no lineales. Dentro del análisis de las cuatro técnicas restantes, se expuso de forma general su algoritmo, dando a conocer el grado de complejidad y de procesamiento de cada uno. Con esta información se elaboró una matriz de alternativas con cuatro criterios elegidos en base a la investigación realizada. A pesar de que dos de las técnicas tienen un proceso similar (PCA y LDA), cada una tiene características que las diferencian en base a los criterios empleados para el análisis. Como resultado de la matriz de alternativas, se seleccionó el método PCA para utilizarlo en la implementación del reconocimiento facial como mecanismo de seguridad para el sistema de soporte en línea propuesto en este proyecto.

## CAPÍTULO III

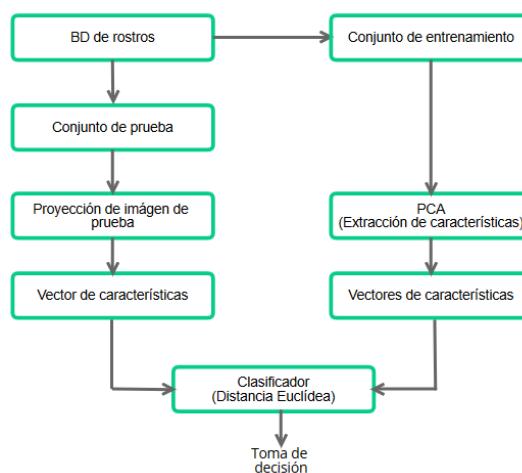
### IMPLEMENTACIÓN DEL RECONOCIMIENTO FACIAL

Para realizar la implementación del reconocimiento facial, se brindará una introducción del algoritmo Eigenfaces el cual, en el capítulo II, tras un análisis de cada algoritmo se seleccionó como el más apropiado para el presente trabajo de grado. Según (Jaiswal, 2011) Eigenfaces es un método para realizar reconocimiento facial basado en un enfoque estadístico. El objetivo de este método es extraer los principales componentes que afectan al máximo la variación de las imágenes. Este es un enfoque holístico, el tratamiento para predecir un rostro se basa en todo el conjunto de entrenamiento.

En este capítulo se realizará la implementación del algoritmo Eigenfaces, se expondrá a fondo sus características, así como su arquitectura y fórmulas, para posteriormente realizar la implementación del algoritmo en el proyecto, con lo cual se dará a conocer paso a paso el desarrollo y los resultados de la implementación con su respectiva conclusión.

#### **3.1. Enfoque PCA para Reconocimiento Facial**

Existen diferentes enfoques para realizar el reconocimiento de un rostro, en los cuales se pueden utilizar métodos estadísticos para encontrar un patrón que representa a una persona en específico (Delbiaggio, 2017). Para este proyecto se ha seleccionado el algoritmo Eigenfaces, anteriormente descrito, para su implementación en el sistema de soporte en línea. Este algoritmo utiliza el método PCA, cuya arquitectura se observa en la Figura 11 y será explicada a continuación.



**Figura 11** Enfoque PCA para reconocimiento facial

Fuente: (Eleyan & Demirel, 2007)

El enfoque PCA se divide en dos fases que son entrenamiento y clasificación, para esto se utiliza una base de datos con conjuntos de rostros para entrenamiento y para prueba. El entrenamiento consiste en la extracción de características de un rostro las cuales formarán un eigenspace que será utilizado en la clasificación. Para la clasificación, una imagen de prueba es proyectada al eigenspace para ser clasificada mediante la distancia euclídea y devolver una respuesta.

### **3.2. Análisis de Componentes Principales (Eigenfaces)**

En este trabajo, las características del rostro se obtienen mediante la implementación del algoritmo de Eigenfaces el cual está basado en el Análisis de Componentes Principales (PCA), este algoritmo según (Lai, Xu, Chen, Yang, & Zhang, 2014), es uno de los más populares de pre procesamiento de datos y extracción de características en los campos de visión por computadora y reconocimiento de patrones.

PCA fue usado por primera vez para representar rostros humanos (Sirovich & Kirby, 1987). Por lo tanto, (Turk & Pentland, 1991) propusieron un famoso método de reconocimiento facial basado en PCA llamado eigenfaces.

Según (Turk & Pentland, 1991) la idea de usar Eigenfaces estuvo motivada por una técnica desarrollada por (Sirovich & Kirby, 1987), para representar imágenes de rostros de manera eficiente mediante el análisis de componentes principales (PCA).

Comenzando con un conjunto de imágenes faciales originales, se calcula un mejor sistema de coordenadas para la comprensión de imágenes, donde cada coordenada es en realidad una imagen que los autores denominan una imagen propia. (Turk & Pentland, 1991) Argumentan que, al menos en principio, cualquier colección de imágenes faciales puede reconstruirse aproximadamente almacenando una pequeña colección de pesos para cada rostro y un pequeño conjunto de imágenes estándar. Los pesos que describen cada rostro se encuentran al proyectar la imagen del rostro en cada foto propia.

Varios autores como (Turk & Pentland, 1991), (Delbiaggio, 2017) y (Slavkovic & Jevtic, 2012) representan el algoritmo con ciertas variaciones pero con un mismo fondo. De acuerdo al análisis realizado, el reconocimiento facial mediante PCA, consta de los siguientes pasos:

### 1) Generar imágenes con las mismas condiciones

Lo primero que se debe hacer es redimensionar las imágenes para que tengan el mismo tamaño en pixeles  $N \times N$  y en caso de ser imágenes a color, deben ser transformadas a escala de grises.

### 2) Formar un conjunto de datos de entrenamiento

En una imagen ( $I_i$ ) 2D cada píxel representa un número. Por lo tanto, pueden ser representados en una matriz  $N \times N$  donde cada elemento de la matriz es un píxel. Si se concatenan las filas de esta matriz, entonces se tendrá la imagen ( $I_i$ ) representada como un vector (Slavkovic & Jevtic, 2012).

$$\mathbf{I} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}_{n \times n} \xrightarrow{\text{CONCATENACIÓN}} \begin{bmatrix} x_{11} \\ \vdots \\ x_{1n} \\ \vdots \\ x_{2n} \\ \vdots \\ x_{mn} \end{bmatrix}_{1 \times N} = \mathbf{x}.$$

**Figura 12** Representación de la imagen  
Fuente: (Slavkovic & Jevtic, 2012)

### 3) Crear una matriz de entrenamiento $X$

Siendo  $M$  el número de vectores imagen, se debe crear una matriz  $X$  de dimensiones  $N \times M$ , de tal modo que:  $X = (x_1, x_2, \dots, x_m)$

### 4) Centrar la matriz $X$

Para asegurar que el primer componente principal describa la dirección de varianza máxima es necesario centrar la matriz  $X$ . Para esto, en primer lugar, se debe determinar el vector de valores promedio.

$$\Psi = \frac{1}{M} \sum_{i=1}^M \mathbf{x}_i \quad (26)$$

Con el vector promedio obtenido, se procede a restar dicho vector de cada vector imagen.

$$\Phi_i = \mathbf{x}_i - \Psi \quad (27)$$

Cada nuevo vector  $\Phi_i$  obtenido genera una nueva matriz de entrenamiento  $A = (\Phi_1, \Phi_2, \dots, \Phi_m)$  de dimensión  $N \times M$ .

### 5) Calcular la matriz de Covarianza $C$ y hallar sus eigenvectors<sup>8</sup> $e_i$ y eigenvalues<sup>9</sup> $\lambda_i$ .

La matriz de covarianza tiene dimensiones  $N \times N$ , por lo tanto, tendrá  $N$  eigenvalues y eigenvectors.

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (28)$$

$$Ce_i = \lambda_i e_i \quad (29)$$

El número de eigenvectors y eigenvalues útiles está limitado por el número de imágenes  $M$  en el conjunto de entrenamiento, es decir que, el cálculo de la matriz  $C$  no sería efectivo.

Uno de los teoremas en álgebra lineal estipula que eigenvectors  $e_i$  y eigenvalues  $\lambda_i$  pueden ser obtenidos encontrando los eigenvectors y eigenvalues de la matriz  $L = A^T A$ , de dimensión  $M \times M$  (Kim, 2008). Si  $v_i$  y  $u_i$  son eigenvectors y eigenvalues de la matriz  $A^T A$ , entonces:

$$A^T A v_i = \mu_i v_i \quad (30)$$

Multiplicando ambos lados de la ecuación (30) por  $A$  en la izquierda, tenemos:

$$\begin{aligned} AA^T A v_i &= A \mu_i v_i, \\ AA^T (Av_i) &= \mu_i (Av_i), \\ C(Av_i) &= \mu_i (Av_i). \end{aligned} \quad (31)$$

Comparando la ecuación (29) y (31), se concluye que los primeros  $M-1$  eigenvectors  $e_i$  y eigenvalues  $\lambda_i$  de la matriz  $C$  están dados por  $Av_i$  y  $u_i$ , respectivamente.

### 6) Formar la matriz de eigenvectors $E$

Se deben ordenar los eigenvectors de mayor a menor de acuerdo a su eigenvalue asociado. Una vez ordenados, se forma la matriz  $E$  de dimensión  $N \times D$ , donde  $D$  representa el número de eigenvectors deseado, de tal forma que cada vector  $e_i$  sea un vector columna en la

<sup>8</sup> Vectores propios de una matriz cuadrada

<sup>9</sup> Valores propios o raíces características de una matriz cuadrada

matriz. La matriz  $E$  se utiliza para la proyección de datos a la matriz  $A$  y para el cálculo de los vectores  $y_i$  de la matriz  $Y = (y_1, \dots, y_m)$ :

$$\mathbf{Y} = \mathbf{E}^T \mathbf{A}. \quad (32)$$

### 7) Reconocimiento de rostros

La imagen de la persona que se desea encontrar es transformada a un vector  $P$ , al cual se le resta el vector promedio para ser proyectada con la matriz de eigenvectores.

$$\omega = \mathbf{E}^T (\mathbf{P} - \Psi). \quad (33)$$

La clasificación se la realiza determinando la distancia  $\varepsilon_i$ , entre  $\omega$  y cada vector  $y_i$  de la matriz  $Y$ . Las distancias más utilizadas son:

1) La distancia Euclídea:

$$d(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^D (a_i - b_i)^2} = \|\mathbf{A} - \mathbf{B}\|. \quad (34)$$

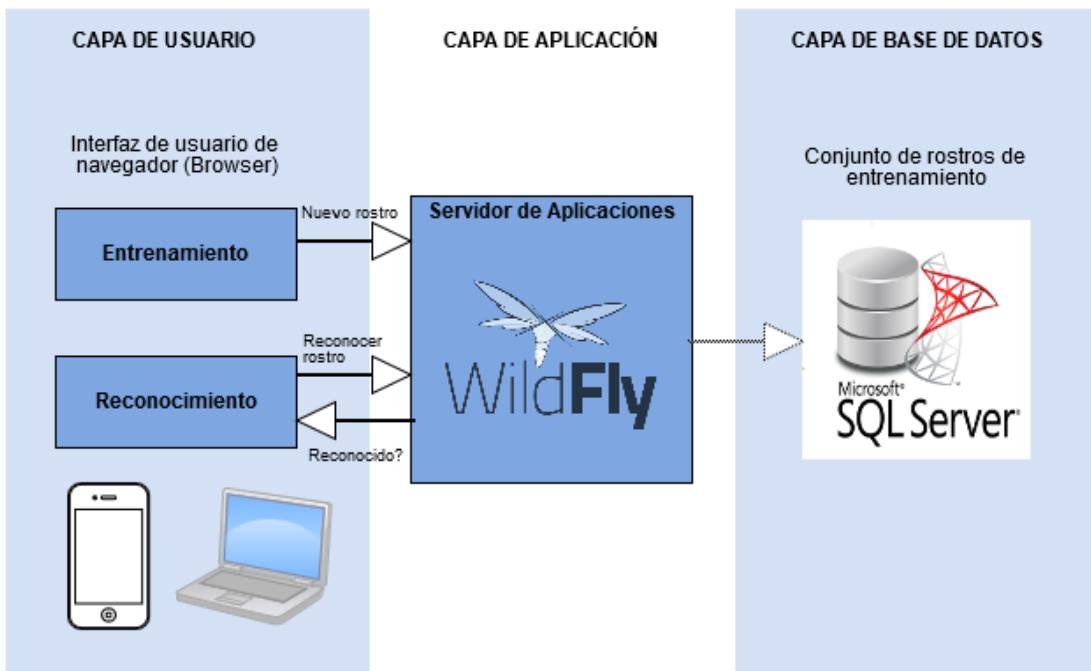
2) La distancia Manhattan:

$$d(\mathbf{A}, \mathbf{B}) = \sum_{i=1}^D |a_i - b_i|; \quad (35)$$

Si la distancia mínima entre la nueva imagen y las imágenes de entrenamiento es mayor que un límite  $\theta$ , la imagen de prueba se considera desconocida, caso contrario es conocida y pertenece a la persona  $s = \arg \min_i [\varepsilon_i]$  (Perlibakas, 2004).

### 3.3. Arquitectura del sistema de reconocimiento facial

El sistema de reconocimiento facial desarrollado en este capítulo aplica los pasos descritos en el [apartado 3.2](#) para entrenar y reconocer rostros. El entrenamiento consiste en agregar un nuevo rostro al conjunto de entrenamiento (Base de datos). Para el reconocimiento se toma una fotografía y se la compara con el conjunto de entrenamiento para encontrar coincidencias. A continuación, se describirá la arquitectura de este sistema:



**Figura 13** Arquitectura de sistema de reconocimiento facial

En primer lugar, la capa de usuario consiste en una interfaz web, esto quiere decir que se requiere de un navegador (browser). Al ser un sistema web, cualquier dispositivo, que tenga un navegador (p. ej. Smartphone, Laptop, Tablet) puede desplegar esta aplicación de reconocimiento facial. El usuario podrá realizar dos funciones, el entrenamiento y el reconocimiento, las cuales harán diferentes peticiones al servidor dentro de la siguiente capa.

En la capa de aplicación se ubica el servidor de aplicaciones. Para este sistema se utilizó WildFly, que puede ser utilizado en cualquier sistema operativo que soporte JAVA, siendo esta una ventaja para la entrega final del producto. Cuando se realiza un entrenamiento, el servidor de aplicaciones se comunica con la base de datos para realizar el registro de la información del nuevo rostro. En cambio, al realizar un reconocimiento, el servidor de aplicaciones sirve de intermediario entre el usuario y la base de datos para saber si el reconocimiento fue exitoso o fallido.

La última capa se refiere a la base de datos en donde se almacena el conjunto de entrenamiento. Pfizer utiliza el gestor de base de datos SQL Server, por lo cual, se utiliza este mismo gestor, tanto para el sistema de reconocimiento facial como para el sistema de soporte en línea que implementará también el reconocimiento facial y que se lo describirá en el siguiente capítulo.

### 3.4. Implementación del algoritmo PCA en JAVA

Con el fin de facilitar la reutilización y optimizar el rendimiento del código desarrollado, se desarrolló una librería con los métodos necesarios para implementar el algoritmo. Los vectores y matrices se manejan por medio de arreglos unidimensionales y bidimensionales respectivamente, por lo cual también se crearon métodos para manipular arreglos. En vista de que el sistema estará basado en la toma de fotografías, también se creó una librería que proporcione los métodos para preparar las imágenes para que pueden ser utilizadas en el proceso de entrenamiento y reconocimiento.

#### 3.4.1. Preparación de la imagen

Para poder utilizar la imagen del rostro, tanto para entrenamiento como para reconocimiento, esta debe ser preparada bajo los criterios analizados en el [apartado 3.2](#). En el siguiente bloque de código, se observa el código desarrollado para la preparación de la imagen:

```
// read image
BufferedImage img = ImageIO.read(cameraFile);
if (cutImage) {
    // cut image to keep the image center
    img = Image.cutImageCenter(img, HORIZONTAL_CUT,
    VERTICAL_CUT);
}
// resize image
img = Image.resizeImage(img, IMG_WIDTH, IMG_HEIGHT, COLOR_TYPE);
if (toGreyScale) {
    // transform image to grey scale
    img = Image.imageToGreyScale(img);
}
return img;
```

**Figura 14** Preparación de la imagen

Este bloque de código, pertenece al método que permite preparar una imagen obtenida de la cámara. La imagen se la procesa en un objeto de la clase *BufferedImage*. Si se requiere cortar la imagen para obtener solo el rostro, se ingresar al método *cutImageCenter*, que recibe la imagen y el porcentaje de la imagen a mantener tanto horizontal como vertical. Con el método *resizeImage*, se redimensiona la imagen, enviándole las nuevas dimensiones y el tipo de color. Esto se hace para que todas las imágenes tengan el mismo tamaño. Por último, si se quiere transformar la imagen a escala de grises, se entra al método *imageToGreyScale*, y se retorna la imagen formateada de acuerdo a las necesidades del algoritmo.

### 3.4.2. Creación del vector imagen

El vector imagen se crea a partir de la imagen obtenida de la preparación. El siguiente bloque de código muestra la creación de este vector:

```
double[] imageVector = new double[IMG_DIMENSION];
// initialize vector values with 0
imageVector = Array.initializeVector(imageVector);
int cont = 0;

for (int x = 0; x < IMG_WIDTH; x++) {
    for (int y = 0; y < IMG_HEIGHT; y++) {
        int p = img.getRGB(x, y);
        // scale the pixel value and assign it the each index
        // of the vector
        imageVector[cont++] =
Number.scaleNumberToPositiveRange(p, OLD_MIN_PXL, OLD_MAX_PXL,
NEW_MAX_PXL);
    }
}

return imageVector;
```

**Figura 15** Creación vector imagen

La parte resaltada corresponde al proceso de creación del vector imagen. Se utilizan dos *for* anidados que van a recorrer cada pixel de la imagen. El valor del pixel, del tipo RGB, en cada iteración, se lo almacena en la variable *p*. Este valor es un entero entre -99999999 a 99999999, por lo cual se lo escala al rango de 0 a 255, con el método *scaleNumberToPositiveRange*. El valor escalado se lo almacena en cada índice del vector imagen y posteriormente, se retorna el mismo.

A partir de este paso, se procede a crear las matrices y vectores que se especifica en el [apartado 3.2](#). Para estos pasos se utilizan los métodos de operación de matrices y vectores mediante arreglos de la librería creada para el proyecto.

### 3.4.3. Obtención de eigenvectors y eigenvalues

Una vez que se deben obtener los vectores y valores propios (eigenvectors y eigenvalues), es necesaria la utilización de una librería de terceros. La librería en cuestión se llama JAMA (Java Matrix Package) y es un paquete de álgebra lineal básica para Java. Principalmente, se utiliza para construir y manipular matrices de gran dimensión.

```

Matrix m = new Matrix(squareMatrix);
EigenvalueDecomposition evd = m.eig();
double[][] matrixEigenvectors = new
double[evd.getV().getRowDimension()][evd.getV().getColumnDimension()];
}

// eigenvectors organized from highest to lowest eigenvalue
int cont = 0, cont2;
for (int i = evd.getV().getRowDimension() - 1; i >= 0; i--) {
    cont2 = 0;
    for (int j = 0; j < evd.getV().getColumnDimension(); j++) {
        matrixEigenvectors[cont2++][cont] = evd.getV().get(j,
i);
    }
    cont++;
}
return matrixEigenvectors;

```

**Figura 16** Obtención de eigenvectores

La Figura 17 muestra la utilización del paquete JAMA para la obtención de los vectores propios. El objeto *m* de la clase *Matrix*, recibe un arreglo bidimensional de  $N \times N$  para su instanciaión. El objeto *evd* de la clase *EigenvalueDecomposition* contiene varios atributos referentes a los eigenvalues de la matriz *m*. Dentro de los *for* anidados se convierte la matriz de eigenvector, obtenido del objeto *evd* con el método *getV*, a una matriz de arreglo bidimensional para mantener el mismo tipo de matrices de todo el proceso.

```

Matrix m = new Matrix(squareMatrix);
EigenvalueDecomposition evd = m.eig();
double[] u = new double[evd.getRealEigenvalues().length];
int cont = 0;
for (int i = evd.getRealEigenvalues().length - 1; i >= 0; i--) {
    u[cont++] = evd.getRealEigenvalues()[i];
}

return u;

```

**Figura 17** Obtención de eigenvalues

La obtención de los eigenvalues es similar a la de los eigenvectores, a diferencia de que con el objeto *evd* de la clase *EigenvaluesDecomposition* se utiliza el método *getRealEigenvalues*. Dicho método retorna un vector con todos los eigenvalues de la matriz *m*, con lo cual dentro del bucle *for* se ordena este vector de mayor a menor valor propio y se lo retorna. Luego de generar los eigenvectores y eigenvalues se continúa con los siguientes pasos determinados en el algoritmo del método PCA.

### 3.5. Ejecución del sistema web de reconocimiento facial

El proceso que sigue el sistema para guardar rostros en la base de datos y reconocer nuevos rostros inicia con la siguiente pantalla:



**Figura 18** Pantalla de inicio RF

En la parte superior, el botón **Cámara** permite realizar la captura del rostro mediante la cámara del dispositivo con el que se utiliza el sistema. El campo **Nombre** sirve para tener una referencia de la persona a la que pertenece la foto tanto al realizar el entrenamiento como al reconocer un nuevo rostro. Los botones **Guardar** y **Reconocer** realizan la función que su nombre indica. Finalmente, las secciones **Foto tomada** y **Foto reconocida** permiten visualizar el rostro con el que se va realizar el entrenamiento o el reconocimiento y el rostro que ha sido reconocido respectivamente.

#### 3.5.1. Guardar rostro (entrenamiento)

El entrenamiento, como ya se mencionó en el capítulo anterior, consiste en guardar las componentes de la imagen en la base de datos. En la Figura 19 se observa una nueva imagen almacenada exitosamente en el conjunto de entrenamiento:



**Figura 19** Entrenamiento

Dentro de la base de datos se realiza el siguiente registro:

PRU_ID	PRU_NOMBRE	PRU_ROSTRO
22	EVELYN ANAGUMB	118.5241886102419,
23	Jonathan Pillajo	119.6147075961470
24	Jefferson Yaguana	117.6853278518532
49	Ejemplo	119.0275050652750
50	Ejemplo	117.6853278518532
26	Rudel Huancas	116.7625810176258,
27	Ivan Conejo	116.5109227901092

**Figura 20** Registro en BD

Se observa que se generó un código para el nuevo registro y el nombre es el que ingresó en el sistema al guarda la imagen. La columna PRU\_ROSTRO contiene una serie de números decimales que corresponden al valor de cada pixel de la fotografía almacenada. Para este sistema las imágenes se redimensionan a una tamaña de 128x150, por lo cual, cada registro contiene 19 200 números decimales en la columna PRU\_ROSTRO.

### 3.5.2. Reconocer un rostro nuevo

Dentro del proceso de reconocimiento se ejecutan los métodos desarrollados de acuerdo al algoritmo de Eigenfaces. La Figura 21 muestra la interfaz de usuario de este proceso:



**Figura 21** Reconocimiento de un rostro nuevo

Al presionar el botón **Reconocer**, la foto tomada es proyectada al conjunto de entrenamiento para compararla con los rostros de la base de datos. En caso de ser exitoso el reconocimiento, la foto de la base de datos considerada como reconocida, se visualiza en la sección **Foto reconocida**, junto con el nombre con el que se registró. Caso contrario, se despliega un mensaje notificando que no se puede reconocer el rostro y la sección **Foto reconocida** queda vacía.

En este capítulo se repasa a detalle el enfoque PCA para reconocimiento facial con el objetivo de dejar claro sus pasos, características, limitantes y su forma de aplicación. Por medio de la revisión de varios autores, se especifica el algoritmo básico que se debe seguir para crear un conjunto de entrenamiento de rostros y posteriormente reconocer un nuevo rostro de entre las imágenes almacenadas. Con el procedimiento definido y comprendido, se procedió a su implementación en Java, dentro de la cual se desarrolló un paquete con los métodos necesarios para el reconocimiento facial y otros métodos utilitarios para manipular imágenes, arreglos y valores numéricos. Dentro de la codificación del algoritmo, se pudo comprobar la afirmación de (Turk & Pentland, 1991), que es posible reconstruir una imagen facial a partir de una colección de pesos de un rostro por medio de un conjunto de imágenes estándar, ya que al obtener el vector de pesos del rostro que fue reconocido, este vector permite generar la imagen del rostro mediante un pequeño proceso inverso del algoritmo, que involucra a la mayor parte

de las imágenes del conjunto de entrenamiento. Finalmente, se puede concluir que, para un reconocimiento exitoso, las imágenes deben ser obtenidas bajo las mismas condiciones de iluminación, calidad de imagen, etc. En el capítulo V se detallarán a fondo estas condiciones por medio de la realización de pruebas.

## CAPÍTULO IV

### DESARROLLO E IMPLEMENTACION DEL SISTEMA DE SOPORTE EN LÍNEA

En este capítulo se realiza la implementación de la aplicación AUTOMATZER. Se expondrá la metodología utilizada, así como los requerimientos funcionales y no funcionales, historias de usuario, arquitectura, diagramas de casos de uso, y modelo de la base de datos. Posteriormente, se realizará la implementación del proyecto, dando a conocer los procesos que se llevaron a cabo para este fin.

Para el desarrollo del sistema de soporte en línea (AUTOMATZER) con el mecanismo de seguridad de reconocimiento facial, se utilizó la metodología de desarrollo ágil SCRUM, que es flexible, ligera, sencilla de entender, y reduce costos y tiempos a la hora del desarrollo del software. Se selecciona esta metodología debido a que es una de las más utilizadas a nivel mundial para el desarrollo de software.

#### **4.1. Metodología SCRUM**

Según (Schwaber & Sutherland, 2013) SCRUM es un marco para el desarrollo, dentro del cual las personas pueden abordar los problemas complejos de adaptación, ya que se basa en la teoría de control del proceso empírico, o el empirismo. SCRUM está conformado por tres pilares esenciales, los cuales son la transparencia, adaptación e inspección.

**Transparencia:** Consiste en que todos los aspectos significativos del proceso, deben ser visibles para el equipo de desarrollo, de modo que comparta un entendimiento de la situación actual del proyecto (Schwaber & Sutherland, 2013).

**Adaptación:** Si en el proceso de desarrollo se determina que uno o más aspectos se desvían de los límites aceptables, se debe realizar un ajuste lo más pronto posible para minimizar el impacto de los errores que pueda ocasionar (Schwaber & Sutherland, 2013).

**Inspección:** El proceso de Inspección, se lo aplica para detectar variaciones en el transcurso del desarrollo, con el fin de alcanzar metas propuestas por el equipo de desarrollo (Schwaber & Sutherland, 2013).

El equipo de SCRUM consiste en un Product Owner, equipo de desarrollo, y un SCRUM Master, a continuación, se mencionarán a cada uno con sus respectivas características.

## **Product Owner**

Es el responsable de dirigir al equipo del desarrollo, así como también de la administración del producto, entre algunas de sus características según (Schwaber & Sutherland, 2013) se encuentran:

- Expresar claramente los ítems del Backlog.
- Ordenar las tareas que se deben realizar en el Backlog para lograr mejores objetivos.
- Designar responsables y tiempos para las tareas expuestas en el Backlog.
- Asegurar que las tareas expuestas en el Backlog se cumplan en el tiempo expuesto.
- Asegurarse que el equipo de desarrollo comprenda las tareas que se expusieron en el Backlog.

## **Equipo de Desarrollo**

Está conformado por profesionales que están facultados por la organización para organizar y gestionar su propio trabajo, en el cual optimiza la eficiencia y la eficacia general del equipo de desarrollo.

Según (Schwaber & Sutherland, 2013) el equipo de desarrollo tiene las siguientes características:

- Son auto-organizados, planifican sus tareas en el tiempo prudente para realizar la entrega del producto.
- Son multifuncionales, con todas las habilidades necesarias para crear un producto de calidad.
- SCRUM no reconoce títulos para los miembros del equipo de desarrollo, es decir unen las características de cada uno y así fomenta el trabajo en equipo.
- Los miembros del Equipo de desarrollo individualmente pueden tener habilidades especializadas y áreas de enfoque, pero la responsabilidad pertenece al Equipo de desarrollo como un todo.

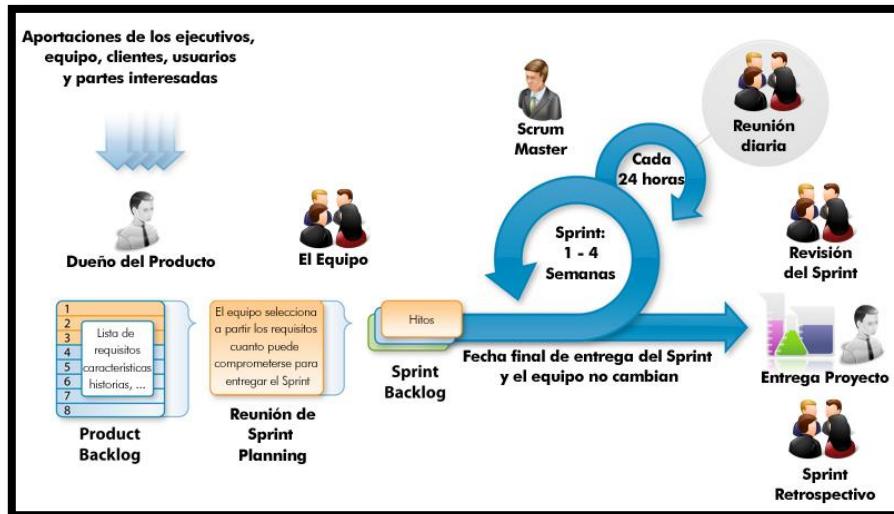
## **Scrum Master**

Es el responsable de que se entienda y promulgue SCRUM, ayudando a los que se encuentran fuera del equipo de Scrum a comprender cuales de sus interacciones con el equipo son útiles y cuáles no.

Según (Schwaber & Sutherland, 2013) algunas de las tareas que realiza el Scrum Master son:

- Encuentra técnicas para una gestión eficaz de las tareas expuestas en el Backlog.
- Comprende la planificación de productos en un entorno empírico.
- Asegura que el Product Owner sepa cómo organizar la acumulación de productos para maximizar el valor.
- Facilita los eventos de SCRUM según lo solicitado o necesario.

La característica esencial de SCRUM es que realiza ciclos parciales lo que permite generar entregables de calidad, en cada SPRINT<sup>10</sup>. Un nuevo Sprint se inicia inmediatamente después de la conclusión del Sprint anterior, y así sucesivamente hasta que el producto esté finalizado. En la Figura 22 se puede visualizar el proceso que realiza SCRUM.



**Figura 22** Metodología SCRUM

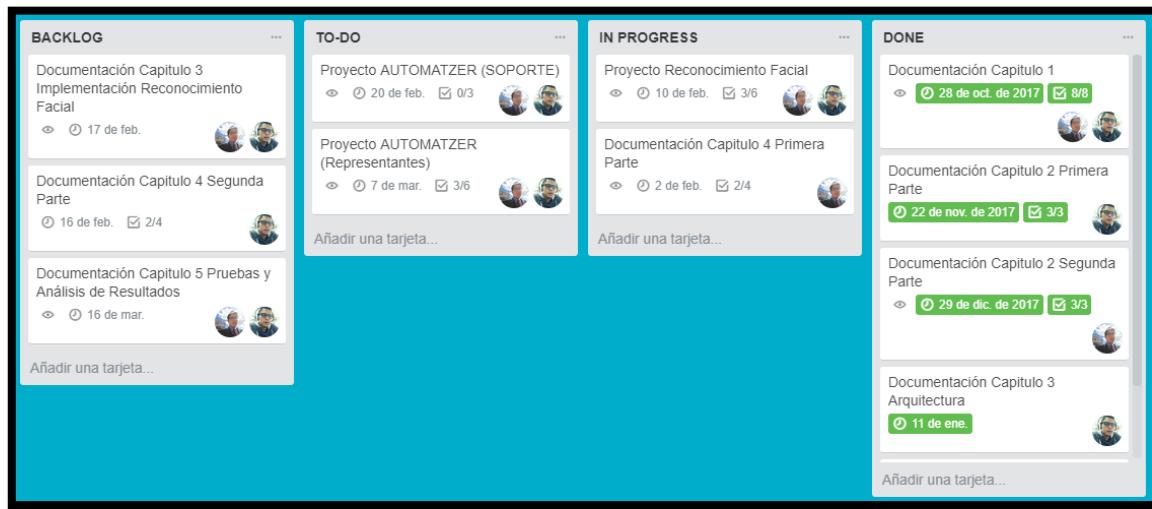
Fuente: (Beedle, 2014)

Para este proyecto de investigación se establecieron a las siguientes personas como miembros del equipo SCRUM, cuyas funciones se detallan a continuación:

- **Product Owner:** Jonathan Pillajo
- **Scrum Master:** Jefferson Yaguana
- **Equipo de Desarrollo:** Jonathan Pillajo, Jefferson Yaguana

<sup>10</sup> Intervalo de tiempo de un mes o menos en la que se desarrollan tareas con la finalidad de entregar un avance utilizable, tangible y funcional.

Una vez que se han establecido los integrantes del equipo SCRUM, y se han asignado las tareas, se realiza el panel de sprint, como se observa en la Figura 23, con las tareas que se debe realizar.



**Figura 23** Panel de Sprint

#### 4.2. Requerimientos Funcionales y No Funcionales

En el presente apartado se presentan los requerimientos funcionales y no funcionales del sistema de soporte en línea. Los requerimientos fueron recolectados por parte del equipo de desarrollo, con ayuda de un representante de Pfizer.

**Tabla 7**  
*Requerimientos Funcionales*

<b>RF01</b>	El sistema tendrá 2 medios de control para el acceso a la información <ul style="list-style-type: none"> <li>○ Credenciales</li> <li>○ Reconocimiento Facial</li> </ul>
<b>RF02</b>	El sistema tendrá 3 perfiles <ul style="list-style-type: none"> <li>○ Usuario Administrador</li> <li>○ Usuario de Soporte</li> <li>○ Usuario Representante</li> </ul>
<b>RF03</b>	El sistema no permitirá el acceso a usuarios que no estén registrados.
<b>RF04</b>	El usuario de soporte podrá realizar escalamientos de solicitudes de ayuda a otros usuarios de soporte.

Continúa →

<b>RF05</b>	El usuario de soporte podrá visualizar el histórico de las solicitudes pendientes.
<b>RF06</b>	El usuario de soporte podrá enviar una respuesta a la solicitud por parte del usuario representante.
<b>RF07</b>	El usuario representante podrá realizar una solicitud para el área de soporte.
<b>RF08</b>	El usuario representante podrá hacer un seguimiento de la solicitud presentada al área de soporte.
<b>RF09</b>	El usuario representante podrá visualizar el histórico de las solicitudes presentadas.
<b>RF10</b>	El sistema mantendrá la persistencia de la información, es decir nunca se eliminará nada de la base de datos.

**Tabla 8**  
*Requerimientos No Funcionales*

<b>RNF01</b>	El sistema deberá contar con una interfaz web amigable para el usuario.
<b>RNF02</b>	El dispositivo tendrá que disponer de acceso a internet para su funcionamiento.
<b>RNF03</b>	El sistema garantiza la seguridad de la información.
<b>RNF04</b>	El sistema estará disponible 24/7
<b>RNF05</b>	El sistema contara con un manual de usuario en español.

Los requerimientos funcionales y no funcionales de la aplicación AUTOMATZER, descritos en la Tabla 7 y la Tabla 8 respectivamente, se especificarán a detalle en el Anexo A. La especificación de estos requerimientos fue elaborada en base al estándar IEEE 830 – 1998, el cual establece directrices para elaborar correctamente una ERS.

#### 4.3. Historias de Usuarios

Una vez analizado los requerimientos funcionales con el personal del área de Soporte SFA de Pfizer, se procedió a realizar las historias de usuario detalladas y que se encuentren acorde a las necesidades planteadas por el área de soporte.

**Nombre del Sistema:** Automatzer

**Descripción:** Automatzer es una aplicación que permitirá al representante de Pfizer, realizar solicitudes de soporte al área de SFA, los mismos que podrán ser respondidos de manera rápida y eficaz, y así se reducir el tiempo de respuesta a la solicitud de estas ayudas. También contará con un sistema de identificación por medio de reconocimiento facial para garantizar la seguridad de la información.

**Tabla 9**

*Historia de Usuario HUPA001*

#### HISTORIA DE USUARIO

<b>Código:</b>	<b>HUPA001</b>	
<b>Módulo:</b>	Login	<b>Rol:</b> Administrador, Soporte, Representante
<b>Descripción corta:</b>	Control de acceso al sistema.	
<b>Prioridad</b>	Alta	
<b>Responsable</b>	Equipo de desarrollo	
<b>Validaciones</b>	<b>Validaciones Generales:</b> <ol style="list-style-type: none"> <li>1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.</li> <li>2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.</li> <li>3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.</li> <li>4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo:</li> </ol>	

Continúa 

- Tamaño máximo del campo es mayor que el permitido.
  - Valor ingresado no corresponde a un número.
  - Valor ingresado contiene caracteres especiales que no son permitidos.
5. El usuario ingresa al sistema siempre y cuando el nombre de usuario y contraseña sean idénticos a los almacenados en la base de datos.

**Descripción:** Pantalla donde se realiza el ingreso del usuario al sistema.

**Restriccione** N/A

S:

**Pantallas:** • Pantalla de login

#### Campos de Entrada

Filtro	Descripción	Longitud	Mandatorio
Nombre de usuario	Nombre del usuario que quiere entrar al sistema	Carácter(100)	Si
Contraseña	Credencial correspondiente el usuario	Carácter(100)	Si Continúa →

---

que quiere  
entrar al  
sistema.

---

#### Acciones (Botones, Links, etc.):

- **Entrar:** Realiza la comprobación de las credenciales para entrar al sistema.
- **Limpiar:** Limpia los campos mostrados en pantalla.

#### Reconocimiento Facial



#### Acciones (Botones, Links, etc.):

- **Tomar Foto:** Abre la cámara del dispositivo para tomar la fotografía que se utiliza para el reconocimiento facial.
- **Regresar:** Regresa a la pantalla de login con credenciales.

Anexos:	N/A
---------	-----

**Tabla 10**  
*Historia de Usuario HUPA002*

<b>HISTORIA DE USUARIO</b>	
<b>Código:</b>	<b>HUPA002</b>
<b>Módulo:</b>	Histórico de solicitudes
<b>Rol:</b>	Administrador, Representante
<b>Descripción corta:</b>	Módulo de soporte para representantes.
<b>Prioridad</b>	Media
<b>Responsable</b>	Equipo de desarrollo
<b>Validaciones:</b>	<p><b>Validaciones Generales:</b></p> <ol style="list-style-type: none"> <li>1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.</li> <li>2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.</li> <li>3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.</li> <li>4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo: <ul style="list-style-type: none"> <li>○ Tamaño máximo del campo es mayor que el permitido.</li> <li>○ Valor ingresado no corresponde a un número.</li> <li>○ Valor ingresado contiene caracteres especiales que no son permitidos.</li> </ul> </li> <li>6. Validar el formato de fechas siempre debe ser “dd-mm-yyyy”.</li> </ol>

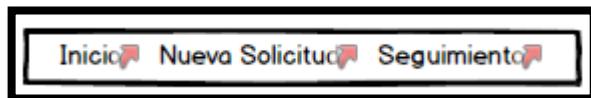
Continúa 

**Descripción:** Pantalla donde se muestra el historial de solicitudes realizadas por el usuario.

**Restricciones:** N/A

**Pantallas:**

- Menú



#### Acciones (Botones, Links, etc.):

- **Inicio:** Despliega la pantalla de inicio (Histórico).
- **Nueva Solicitud:** Despliega la pantalla para crear una solicitud de ayuda.
- **Seguimiento:** Despliega la pantalla de seguimiento de las solicitudes realizadas.

#### • Pantalla de inicio

Imagen de la pantalla de inicio que muestra una lista de solicitudes realizadas y un cuadro de diálogo para ver más detalles.

Solicitudes realizadas		
Nombre (solicitud)	Fecha	Estado
S1	02/05/2017	<input type="radio"/>
S2	02/05/2017	<input checked="" type="checkbox"/>
S3	02/05/2017	<input type="checkbox"/>
S4	02/05/2017	<input checked="" type="checkbox"/>

Detalle de solicitud...  
S1...

#### Detalle de los campos de la grilla:

**Nombre (solicitud):** Nombre de la solicitud.

**Fecha:** Fecha en que se generó la solicitud.

**Estado:** Estado en el que se encuentra la solicitud.

**Campo de Texto:** Descripción de la solicitud.

**Anexos:** N/A

**Tabla 11**  
*Historia de Usuario HUPA003*

<b>HISTORIA DE USUARIO</b>	
<b>Código:</b>	<b>HUPA003</b>
<b>Módulo:</b>	Creación de una nueva solicitud
<b>Rol:</b>	Administrador, Representante
<b>Descripción corta:</b>	Módulo de soporte para representantes.
<b>Prioridad</b>	Media
<b>Responsable</b>	Equipo de desarrollo
<b>Validaciones:</b>	
<b>Validaciones Generales:</b>	
<ol style="list-style-type: none"> <li>1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.</li> <li>2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.</li> <li>3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.</li> <li>4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo: <ul style="list-style-type: none"> <li>○ Tamaño máximo del campo es mayor que el permitido.</li> <li>○ Valor ingresado no corresponde a un número.</li> <li>○ Valor ingresado contiene caracteres especiales que no son permitidos.</li> </ul> </li> </ol>	

Continúa 

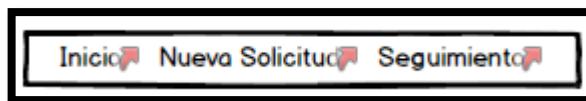
5. Validar el formato de fechas siempre debe ser “dd-mm-yyyy”.
6. Validar formato de campos [Decimales: utilizar "punto"; Número: miles utilizar "coma"].
7. Revisión de inconsistencias entre campos dependientes, ejemplo: combos anidados, rango de fechas.

**Descripción:** Pantalla donde se muestra el historial de solicitudes realizadas por el usuario.

**Restricciones:** N/A

**Pantallas:**

- Menú



#### Acciones (Botones, Links, etc.):

- **Inicio:** Despliega la pantalla de inicio (Histórico).
  - **Nueva Solicitud:** Despliega la pantalla para crear una solicitud de ayuda.
  - **Seguimiento:** Despliega la pantalla de seguimiento de las solicitudes realizadas.
- **Pantalla de nueva solicitud**

#### Campos de Entrada

Filtro	Descripción	Longitud	Mandatorio
<b>Tipo de solicitud</b>	Nombre de la solicitud que se va a realizar.	Carácter(100)	Si
<b>Detalle solicitud</b>	Descripción adicional para la solicitud que se va a realizar.	Carácter(500)	No

**Acciones (Botones, Links, etc.):**

- **Guardar:** Guarda la solicitud.
- **Guardar y Enviar:** Guarda la solicitud y la envía para que sea resulta.
- **Cancelar:** Limpia los campos llenados de la nueva solicitud.

<b>Anexos:</b>	N/A
----------------	-----

**Tabla 12***Historia de Usuario HUPA004***HISTORIA DE USUARIO**

<b>Código:</b>	<b>HUPA004</b>
<b>Módulo:</b>	Seguimiento de Solicituds
<b>Descripción corta:</b>	Módulo de soporte para representantes.
<b>Prioridad</b>	Media
<b>Responsable</b>	Equipo de desarrollo
<b>Validaciones:</b>	<p><b>Validaciones Generales:</b></p> <ol style="list-style-type: none"> <li>1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.</li> </ol>

Continúa 

2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (\*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.
3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.
4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo:
  - Tamaño máximo del campo es mayor que el permitido.
  - Valor ingresado no corresponde a un número.
  - Valor ingresado contiene caracteres especiales que no son permitidos.
5. Validar el formato de fechas siempre debe ser “dd-mm-yyyy”.
6. Validar formato de campos [Decimales: utilizar "punto"; Número: miles utilizar "coma"].
7. Revisión de inconsistencias entre campos dependientes, ejemplo: combos anidados, rango de fechas.

**Descripción:** Pantalla donde se realiza el seguimiento de las solicitudes enviadas al área de soporte.

**Restricciones:** N/A

**Pantallas:**

- Menú



#### Acciones (Botones, Links, etc.):

- **Inicio:** Despliega la pantalla de inicio (Histórico).
- **Nueva Solicitud:** Despliega la pantalla para crear una solicitud de ayuda.

Continúa →

- **Seguimiento:** Despliega la pantalla de seguimiento de las solicitudes realizadas.
- **Pantalla de seguimiento de solicitud**

Seguimiento de solicitudes		
Nombre (solicitud)	Fecha	Estado
S1	02/05/2017	<input type="radio"/>
S2	02/05/2017	<input checked="" type="checkbox"/>
S3	02/05/2017	<input type="checkbox"/>
S4	02/05/2017	<input checked="" type="checkbox"/>

Respuesta 1  
Respuesta 2  
Respuesta 3  
..  
..

#### Detalle de los campos de la grilla:

**Nombre (solicitud):** Nombre de la solicitud.

**Fecha:** Fecha en que se generó la solicitud.

**Estado:** Estado en el que se encuentra la solicitud.

**Campo de Texto:** Respuestas generadas para la solicitud seleccionada.

**Anexos:** N/A

**Tabla 13**  
*Historia de Usuario HUPA005*

#### HISTORIA DE USUARIO

<b>Código:</b>	<b>HUPA005</b>	
<b>Módulo:</b>	Histórico de solicitudes	Rol: Administrador, Soporte
<b>Descripción corta:</b>	Módulo de atención de soporte.	
<b>Prioridad</b>	Media	
<b>Responsable</b>	Equipo de desarrollo	
<b>Validaciones:</b>	<b>Validaciones Generales:</b> <span style="float: right;">Continúa →</span>	

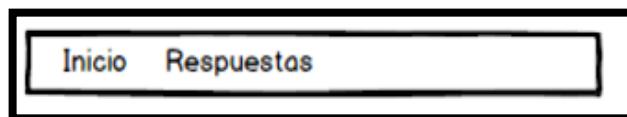
1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.
2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (\*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.
3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.
4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo:
  - Tamaño máximo del campo es mayor que el permitido.
  - Valor ingresado no corresponde a un número.
  - Valor ingresado contiene caracteres especiales que no son permitidos.
5. Validar el formato de fechas siempre debe ser “dd-mm-yyyy”.

**Descripción:** Pantalla donde se muestra el historial de solicitudes registradas.

**Restricciones:** N/A

**Pantallas:**

- Menú



#### Acciones (Botones, Links, etc.):

- **Inicio:** Despliega la pantalla de inicio (Histórico).
- **Respuesta:** Despliega la pantalla para dar una respuesta a la solicitud de los representantes. **Continúa** →

- **Pantalla de inicio**

Solicitudes registradas

Nombre (solicitud)	Fecha	Estado
S1	02/05/2017	<input type="radio"/>
S2	02/05/2017	<input checked="" type="checkbox"/>
S3	02/05/2017	<input type="checkbox"/>
S4	02/05/2017	<input checked="" type="checkbox"/>

Detalle de solicitud...  
S1...

#### Detalle de los campos de la grilla:

**Nombre (solicitud):** Nombre de la solicitud.

**Fecha:** Fecha en que se generó la solicitud.

**Estado:** Estado en el que se encuentra la solicitud.

**Campo de Texto:** Descripción de la solicitud.

#### **Botones:**

- **Resolver:** Marca como resuelta la solicitud.
- **Escalar:** Despliega la pantalla para poder enviar un correo a la siguiente persona la cual puede resolver la solicitud.

**Anexos:** N/A

**Tabla 14**

*Historia de Usuario HUPA006*

#### **HISTORIA DE USUARIO**

**Código:** **HUPA007**

**Módulo:** Escalar                   **Rol:** Administrador, Soporte  
solicitud

**Descripción corta:** Módulo de atención de soporte.

**Prioridad** Media

Continúa →

<b>Responsable</b>	Equipo de desarrollo
<b>Validaciones:</b>	
<b>Validaciones Generales:</b>	
	<ol style="list-style-type: none"> <li>1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.</li> <li>2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.</li> <li>3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.</li> <li>4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo: <ul style="list-style-type: none"> <li>○ Tamaño máximo del campo es mayor que el permitido.</li> <li>○ Valor ingresado no corresponde a un número.</li> <li>○ Valor ingresado contiene caracteres especiales que no son permitidos.</li> </ul> </li> <li>5. Validar el formato de fechas siempre debe ser “dd-mm-yyyy”.</li> <li>6. Si no se llenan los campos de asunto y mensaje no se podrá continuar con el escalamiento</li> </ol>
<b>Descripción:</b>	Pantalla que permite realizar un escalamiento a otro personal del área de soporte, debido a que no se puede realizar la solicitud. <b>Continúa</b> 
<b>Restricciones:</b>	N/A
<b>Pantallas:</b>	<ul style="list-style-type: none"> <li>• Menú</li> </ul> <div style="border: 1px solid black; padding: 10px; text-align: center;">  </div>

### Acciones (Botones, Links, etc.):

- **Inicio:** Despliega la pantalla de inicio (Histórico).
- **Respuesta:** Despliega la pantalla para dar una respuesta a la solicitud de los representantes.
- **Pantalla de Escalar**

The screenshot shows a window titled "Escalar solicitud". At the top, there are three input fields: "Nombre (solicitud)" with value "S1", "Fecha" with value "02/05/2017", and "Estado". Below this is a section titled "Correo" containing "Asunto:" and "Mensaje:" fields. The "Mensaje:" field contains the text "Detalle de solicitud... S1...". At the bottom are two buttons: "Enviar" and "Cancelar".

### Campos de Entrada

Filtro	Descripción	Longitud	Mandatorio
<b>Asunto</b>	Asunto que se le da al mensaje de escalamiento.	Carácter(100)	Si
<b>Mensaje</b>	Detalla el motivo por el cual se realiza la escala a la siguiente persona.	Carácter(500)	Si

**Continúa** ➔

### Detalle de los campos de la grilla:

**Nombre (solicitud):** Nombre de la solicitud.

**Fecha:** Fecha en que se generó la solicitud.

**Estado:** Estado en el que se encuentra la solicitud.

**Botones:**

- **Enviar:** Envía el Escalamiento.
- **Cancelar:** Cancela la ejecución de la escala de la solicitud.

<b>Anexos:</b>	N/A
----------------	-----

**Tabla 15**  
*Historia de Usuario HUPA007*

**HISTORIA DE USUARIO**

<b>Código:</b>	<b>HUPA007</b>	
<b>Módulo:</b>	Solicitudes	Rol: Administrador, Soporte registradas
<b>Descripción corta:</b>	Módulo de atención de soporte.	
<b>Prioridad</b>	Media	
<b>Responsable</b>	Equipo de desarrollo	
<b>Validaciones:</b>	<p><b>Validaciones Generales:</b></p> <ol style="list-style-type: none"> <li>1. Validar que se cumplan los formatos de cada campo tanto en longitud como en tipo y restringir el ingreso de más caracteres.</li> <li>2. Validar que los campos que son considerados obligatorios sean solicitados por el sistema, los cuales se deben encontrar marcados con un asterisco en la pantalla (*), en caso de no ingresar este valor obligatorio el sistema debe mostrar el mensaje de error: “Campo Obligatorio”.</li> <li>3. En la búsqueda el uso de mayúsculas o minúsculas sea indistinto.</li> <li>4. Si el usuario ingresa un valor no válido el sistema emite un mensaje parametrizado en que se indica el por qué el valor no es válido. Por ejemplo:</li> </ol>	

**Continúa** 

- Tamaño máximo del campo es mayor que el permitido.
  - Valor ingresado no corresponde a un número.
  - Valor ingresado contiene caracteres especiales que no son permitidos.
5. Validar el formato de fechas siempre debe ser “dd-mm-yyyy”.
  6. Si no se llena el campo de respuesta, no se podrá dar como terminada la solicitud.

**Descripción:** Pantalla que permite visualizar las solicitudes pendientes que posee el área de soporte, las cuales se ordenan en forma descendente por el parámetro fecha.

**Restricciones:** N/A

**Pantallas:**

- Menú



**Acciones (Botones, Links, etc.):**

- **Inicio:** Despliega la pantalla de inicio (Histórico).
- **Respuesta:** Despliega la pantalla para dar una respuesta a la solicitud de los representantes.
- **Pantalla para Responder la solicitud**

Continúa ➔

Solicitudes registradas

Nombre (solicitud)	Fecha	Estado
S1	02/05/2017	<input type="radio"/>
S2	02/05/2017	<input checked="" type="checkbox"/>
S3	02/05/2017	<input type="checkbox"/>
S4	02/05/2017	<input checked="" type="checkbox"/>

Solicitud aceptada...

Responder

#### Detalle de los campos de la grilla:

**Nombre (solicitud):** Nombre de la solicitud.

**Fecha:** Fecha en que se generó la solicitud.

**Estado:** Estado en el que se encuentra la solicitud.

**Campo de Texto:** Detallar una respuesta a la solicitud por parte de los representantes

#### **Botones:**

- **Responder:** Envía el detalle que se encuentra en el campo de texto al representante el cual realizó la solicitud.

**Anexos:**

N/A

#### 4.4. Modelo físico de base de datos

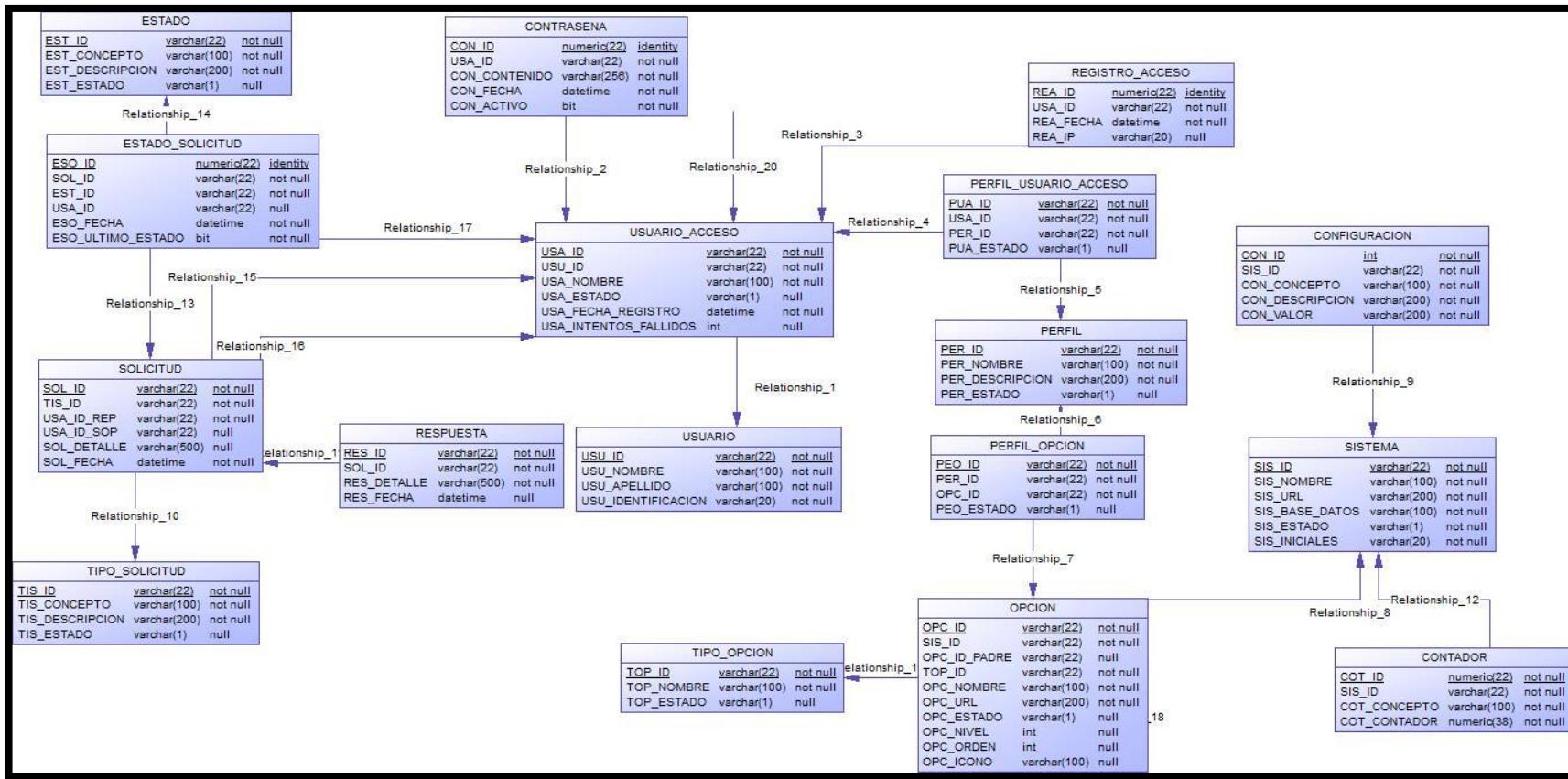


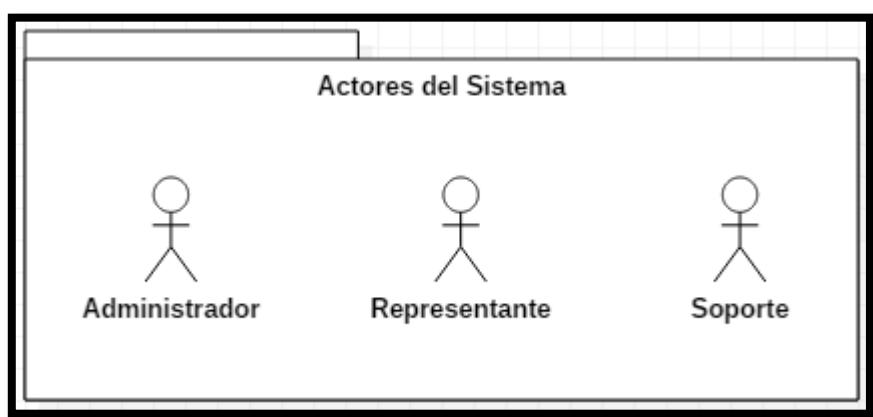
Figura 24 Modelo físico de la base de datos AUTOMATZER

El Modelo lógico de la base de datos AUTOMATZER se la encontrará como Anexo B y el diccionario de datos como Anexo C.

## 4.5. Diagramas de Casos de Uso

### 4.5.1. Definición de Actores del Sistema

Previo al desarrollo de la aplicación Automatzer, se procedió a definir los actores que se encuentran involucrados en el sistema: a) usuario administrador quien se encarga de todas las actualizaciones que requiera el sistema, además de que tiene acceso a todos los procesos del sistema, b) usuario representante es aquel que realiza las peticiones de solicitud al área de soporte y c) usuario de soporte es el encargado de dar solución a la solicitud que envía el usuario representante. Se puede observar en la Figura 25



*Figura 25* Definición de Actores StarUML

### 4.5.2. Definición de tareas por Actor

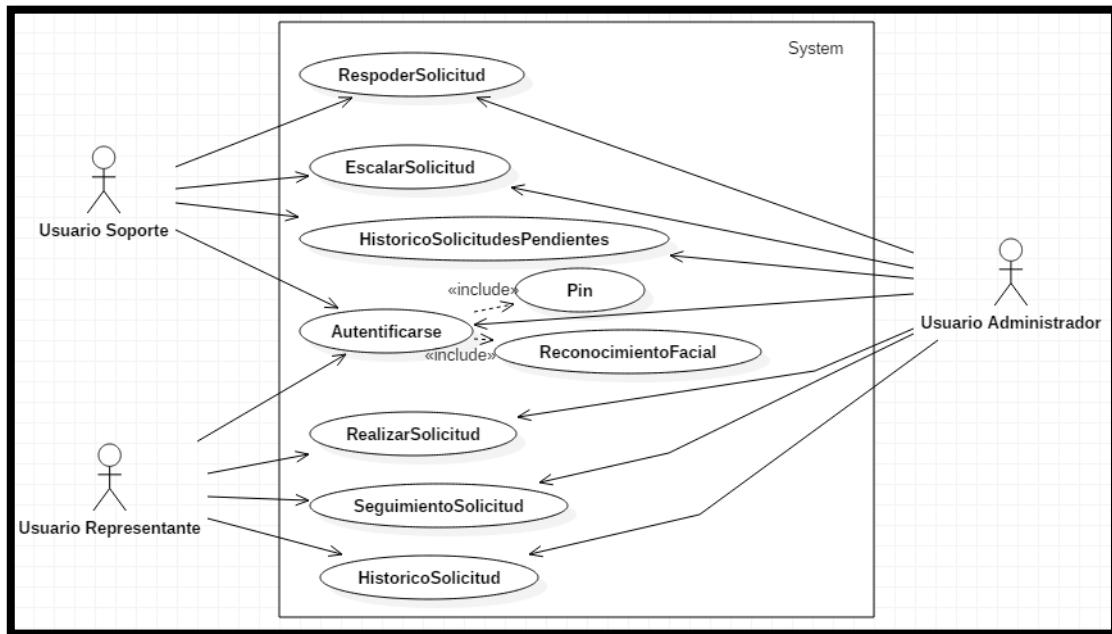
Las tareas que realiza cada usuario en el sistema se podrá observar de forma más detallada en la Figura 26.

Tareas \ Usuarios	Autentificarse	Historico de Solicitudes	Realizar Solicitud	Realizar Seguimiento de Solicitudes	Responder Solicitud	Escalar Solicitud	Historico de Solicitudes Atendidas
Administrador	<input checked="" type="checkbox"/>						
Representante	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Soporte	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

*Figura 26* Tareas por actor

#### 4.5.3. Definición de casos de uso general

Uno de los objetivos de los casos de uso es entender como los usuarios interactúan con el sistema (ver Figura 27)



**Figura 27** Casos de Uso General StarUML

#### 4.5.4. Descripción de casos de uso

A continuación, se detallará cada caso de uso para tener un mejor entendimiento del sistema.

**Tabla 16**

*Caso de Uso CU\_001 “Autentificarse”*

Caso de uso:	<b>Autentificarse (CU_001)</b>
Descripción:	Permite a los usuarios registrados acceder a la aplicación
Actores:	Usuario administrador, Usuario Representante, Usuario Soporte
Pre-condiciones:	El usuario debe estar registrado en el sistema
Actividades:	
Escenario principal:	a) Pin 1. El usuario ingresa su nombre de usuario y contraseña.

Continúa →

	<ol style="list-style-type: none"> <li>2. El usuario da clic en el botón “Ingresar”.</li> <li>3. El sistema verifica que la información ha sido ingresada correctamente, y que el usuario este previamente registrado en el sistema.</li> <li>4. El sistema muestra la pantalla principal de la aplicación, donde se muestran las acciones que el usuario puede realizar dependiendo del perfil que este asignado.</li> </ol> <p>b) Reconocimiento Facial</p> <ol style="list-style-type: none"> <li>1. El usuario se toma una foto con la cámara de su dispositivo</li> <li>2. El Usuario da clic en el botón “Ingresar”</li> <li>3. El sistema verifica mediante el algoritmo EigenFaces si se encuentra registrado.</li> <li>4. El sistema muestra la pantalla principal de la aplicación, donde se muestran las acciones que el usuario puede realizar dependiendo del perfil que este asignado.</li> </ol>
Escenario alterno:	<p>a. Los datos de ingreso no son correctos</p> <ol style="list-style-type: none"> <li>1. El sistema muestra mensaje de error de datos incorrectos.</li> <li>2. El usuario corrige los datos de ingreso.</li> <li>3. Continúa con el punto 2 del escenario principal.</li> </ol>

**Tabla 17***Caso de Uso CU\_002 “Realizar Solicitud”*

Caso de uso:	<b>Realizar Solicitud (CU_002)</b>
Descripción:	Permite a los usuarios enviar una solicitud al área de soporte de Pfizer
Actores:	Usuario administrador, Usuario Representante
Pre-condiciones:	El usuario debe estar registrado en el sistema
Actividades:	

Continúa 

Escenario principal:	<ol style="list-style-type: none"> <li>5. El usuario se dirige a la opción de realizar solicitud.</li> <li>6. Llena los campos tipo de solicitud y el detalle de solicitud.</li> <li>7. El usuario da clic en el botón “guardar y enviar”</li> <li>8. El sistema verifica que la información ha sido ingresada correctamente.</li> <li>9. El sistema muestra un mensaje de “solicitud enviada correctamente”</li> </ol>
Escenario alterno:	<p>a) Los datos de ingreso no son correctos</p> <ol style="list-style-type: none"> <li>4. El sistema muestra mensaje de error de datos incorrectos.</li> <li>5. El usuario corrige los datos de ingreso.</li> <li>6. Continúa con el punto 3 del escenario principal.</li> </ol> <p>b) El usuario pulsa el botón cancelar</p> <ol style="list-style-type: none"> <li>1. El sistema limpiara todos los campos de la pantalla.</li> <li>c) El usuario pulsa el botón guardar</li> <ol style="list-style-type: none"> <li>1. El sistema verifica que la información ha sido ingresada correctamente.</li> <li>2. El sistema almacena en la base de datos la información que ha sido ingresada pero no envía al área de soporte.</li> </ol> </ol>

**Tabla 18***Caso de Uso CU\_003 “Seguimiento Solicitud”*

Caso de uso:	<b>Seguimiento Solicitud (CU_003)</b>
Descripción:	Permite a los usuarios hacer un seguimiento a las solicitudes que hayan enviado al área de soporte.
Actores:	Usuario administrador, Usuario Representante
Pre-condiciones:	El usuario debe estar registrado en el sistema

Continúa 

Actividades:

Escenario principal:	<ol style="list-style-type: none"> <li>1. El usuario se dirige a la opción de seguimiento de solicitud.</li> <li>2. El sistema le permitirá visualizar las solicitudes que aún no han sido atendidas</li> <li>3. El usuario seleccionara la solicitud que le quiera dar seguimiento.</li> <li>4. El usuario una vez seleccionada la solicitud, dará clic en el botón “Seguimiento”</li> <li>5. El sistema le mostrara una pantalla con el estado, la fecha en la que se ha enviado la solicitud y si ya ha sido recibida por un usuario de soporte.</li> </ol>
Escenario alterno:	<ol style="list-style-type: none"> <li>1. Si no existe ninguna solicitud pendiente, el sistema mostrará un mensaje que diga “No existen solicitudes para dar seguimiento”</li> </ol>

**Tabla 19**

Caso de Uso CU\_004 “Histórico Solicitud”

Caso de uso:	<b>Histórico Solicitud (CU_004)</b>
Descripción:	Permite a los usuarios visualizar las solicitudes que han enviado al área de soporte.
Actores:	Usuario administrador, Usuario Representante
Pre-condiciones:	El usuario debe estar registrado en el sistema
Actividades:	<p>Escenario principal:</p> <ol style="list-style-type: none"> <li>1. El sistema le permitirá visualizar las solicitudes que el usuario a enviado al área de soporte</li> <li>2. El usuario podrá ordenar las solicitudes por nombre, fecha y estado.</li> </ol>

Continúa 

Escenario alterno:	1. Si no existe ninguna solicitud el sistema mostrará un mensaje que diga “No existen solicitudes enviadas”
--------------------	---

**Tabla 20***Caso de Uso CU\_005 “Responder Solicitud”*

Caso de uso:	<b>Responder Solicitud (CU_005)</b>
Descripción:	Permite a los usuarios responder una solicitud enviado por el representante.
Actores:	Usuario administrador, Usuario Soporte
Pre-condiciones:	El usuario debe estar registrado en el sistema
Actividades:	<p>Escenario principal:</p> <ol style="list-style-type: none"> <li>1. El usuario se dirige a la opción de responder solicitud.</li> <li>2. El sistema permitirá visualizar al usuario las solicitudes que le han llegado.</li> <li>3. El usuario selecciona una solicitud.</li> <li>4. El usuario luego de resolver la solicitud en el campo de detalle de solicitud, explicara lo que necesito para resolverla.</li> <li>5. El usuario da clic en el botón “Resolver”</li> <li>6. El sistema verifica que la información ha sido ingresada correctamente.</li> <li>7. El sistema muestra un mensaje de “solicitud resuelta correctamente”</li> </ol> <p>Escenario alterno:</p> <ul style="list-style-type: none"> <li>a) Los datos de ingreso no son correctos <ul style="list-style-type: none"> <li>1. El sistema muestra mensaje de error de datos incorrectos.</li> <li>2. El usuario corrige los datos de ingreso.</li> <li>3. Continúa con el punto 3 del escenario principal.</li> </ul> </li> <li>b) El usuario pulsa el botón cancelar</li> </ul>

Continúa →

- 
1. El sistema limpiara todos los campos de la pantalla.
- 

**Tabla 21***Caso de Uso CU\_006 “Escalar Solicitud”*

Caso de uso:	<b>Escalar Solicitud (CU_006)</b>
Descripción:	Permite a los usuarios hacer un escalamiento de una solicitud a otro usuario de soporte.
Actores:	Usuario administrador, Usuario Soporte
Pre-condiciones:	El usuario debe estar registrado en el sistema
Actividades:	
Escenario principal:	<ol style="list-style-type: none"> <li>1. El usuario se dirige a la opción de escalar solicitud.</li> <li>2. El sistema le permitirá visualizar las solicitudes que aún no han sido atendidas</li> <li>3. El usuario seleccionara la solicitud que le quiera escalar.</li> <li>4. El usuario una vez seleccionada la solicitud, llenara los campos de asunto y mensaje.</li> <li>5. El usuario dará clic en el botón “Enviar”</li> <li>6. El sistema validara que todos los campos se hayan llenado correctamente y mostrara un mensaje de “El escalamiento se ha realizado correctamente”</li> </ol>
Escenario alterno:	<ol style="list-style-type: none"> <li>1. Si no existe ninguna solicitud pendiente, el sistema mostrará un mensaje que diga “No existen solicitudes para realizar el escalamiento”.</li> <li>2. Si el usuario da clic en el botón “Cancelar” se limpiará todos los campos.</li> <li>3. Si no le llenaron los campos correctamente el sistema mostrara un mensaje de error que diga</li> </ol>

Continúa 

“Llene los campos necesarios para realizar el escalamiento.

**Tabla 22***Caso de Uso CU\_007 “Histórico Solicitud Pendiente”*

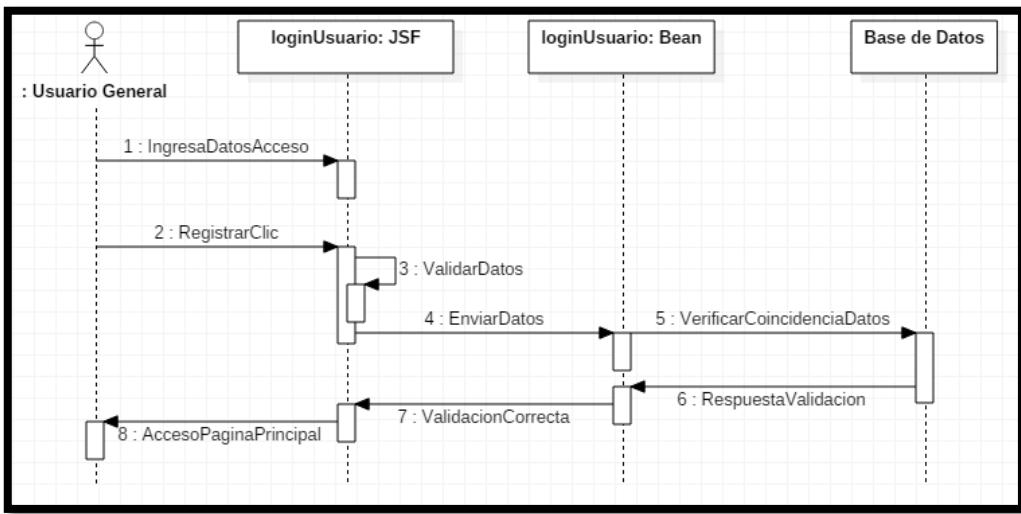
Caso de uso:	<b>Histórico Solicitud Pendiente (CU_007)</b>
Descripción:	Permite a los usuarios visualizar las solicitudes que han recibido el área de soporte por parte de los representantes.
Actores:	Usuario administrador, Usuario Soporte
Pre-condiciones:	El usuario debe estar registrado en el sistema
Actividades:	<p>Escenario principal:</p> <ol style="list-style-type: none"> <li>1. El sistema le permitirá visualizar las solicitudes que el usuario de soporte a recibido por parte de los representantes.</li> <li>2. El usuario podrá ordenar las solicitudes por nombre, fecha y estado.</li> </ol>
Escenario alterno:	<ol style="list-style-type: none"> <li>1. Si no existe ninguna solicitud el sistema mostrará un mensaje que diga “No existen solicitudes recibidas”</li> </ol>

## 4.6. Diagramas de Secuencia

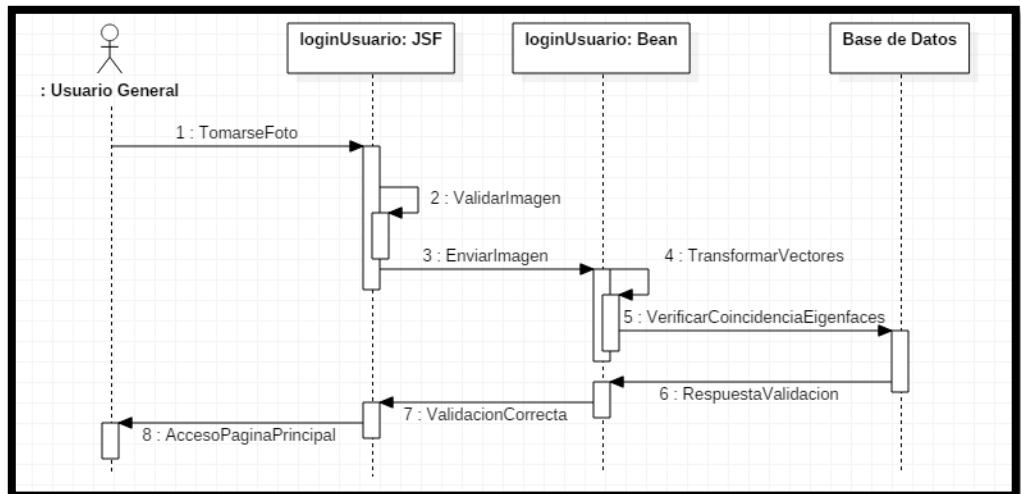
A continuación, se dará a conocer los diagramas de secuencia del sistema AUTOMATZER.

### 4.6.1. Diagrama de Secuencia “Autentificación de Usuario”

En este caso existirán dos diagramas de secuencia diferentes, ya que existen dos formas de autentificación que son por medio de pin (usuario y clave) que se muestra en la Figura 28, y por medio de reconocimiento facial que se muestra en la Figura 29.



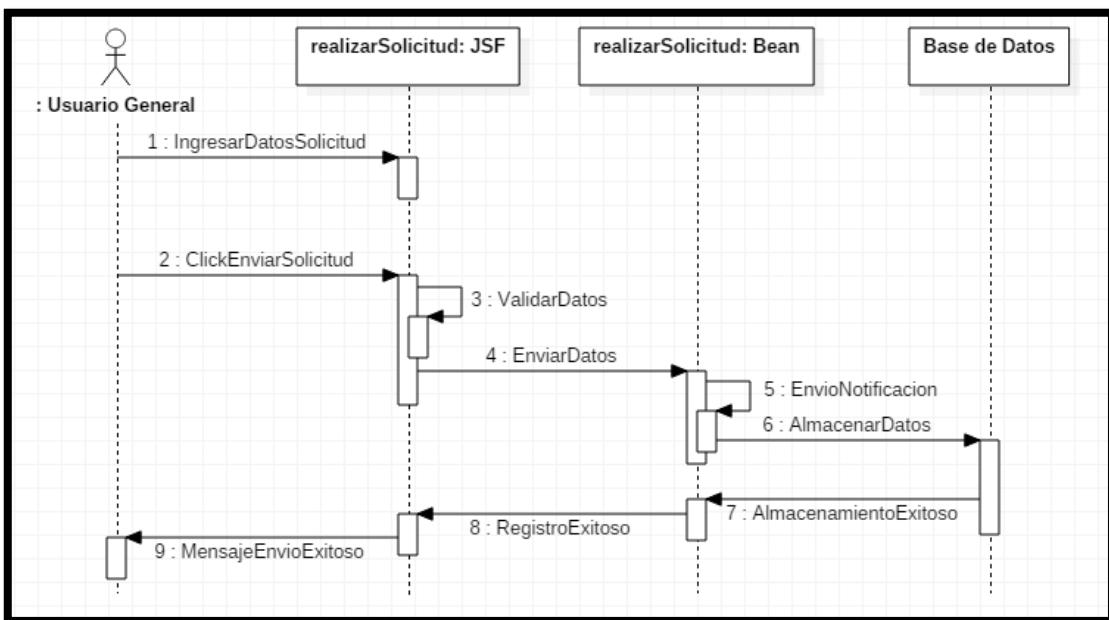
**Figura 28** Diagrama de Secuencia Autentificación Usuario (Pin)



**Figura 29** Diagrama de Secuencia Autentificación Usuario (Reconocimiento Facial)

#### 4.6.2. Diagrama de Secuencia “Realizar Solicitud”

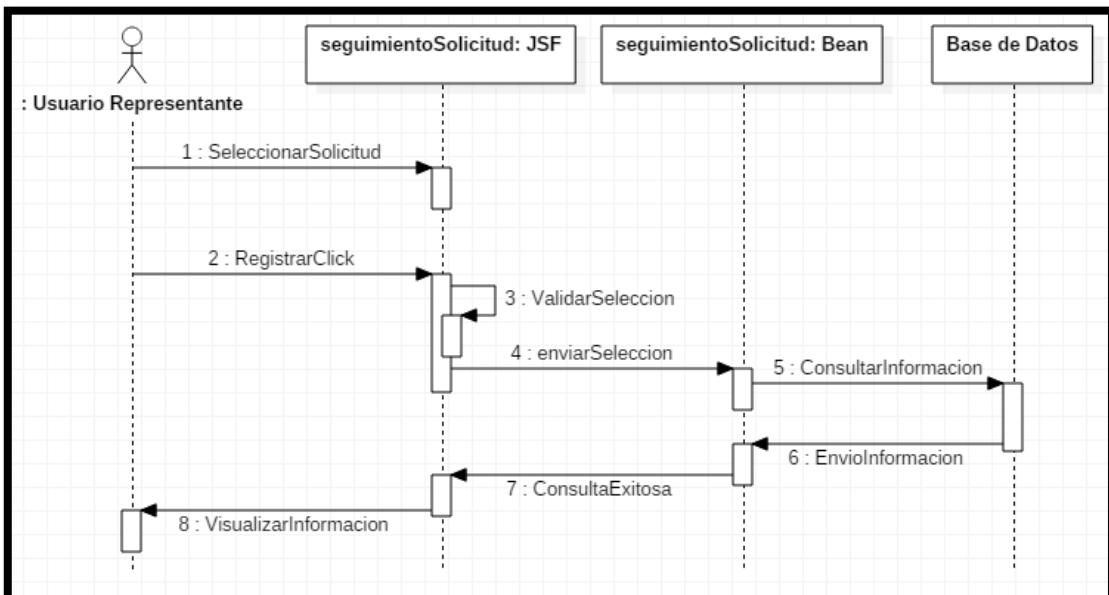
En la figura 30 se muestra el diagrama de secuencia para realizar una solicitud al área de soporte.



**Figura 30** Diagrama de Secuencia Realizar Solicitud

#### 4.6.3. Diagrama de Secuencia “Seguimiento Solicitud”

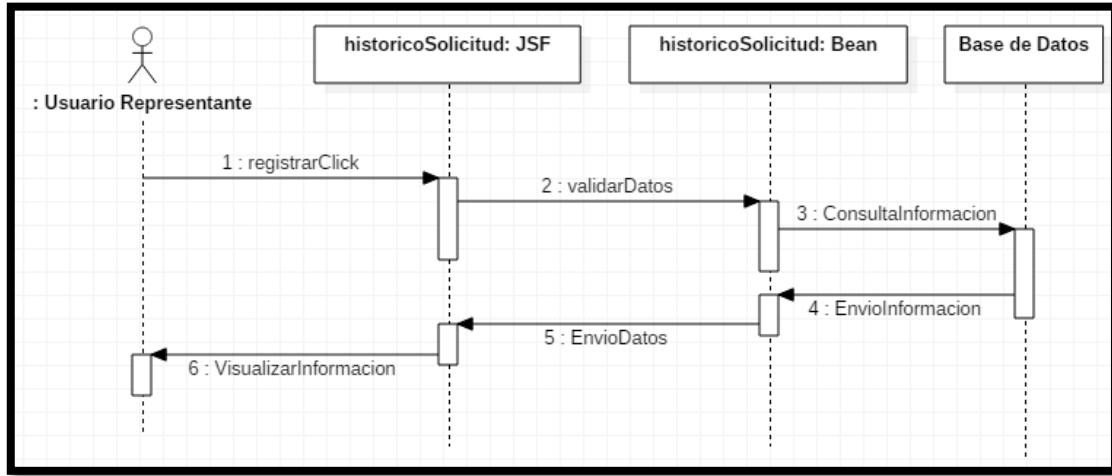
En la Figura 31 se visualiza el diagrama de secuencia para dar seguimiento a una solicitud realizada al área de soporte.



**Figura 31** Diagrama de Secuencia Seguimiento Solicitud

#### 4.6.4. Diagrama de Secuencia “Histórico Solicitud”

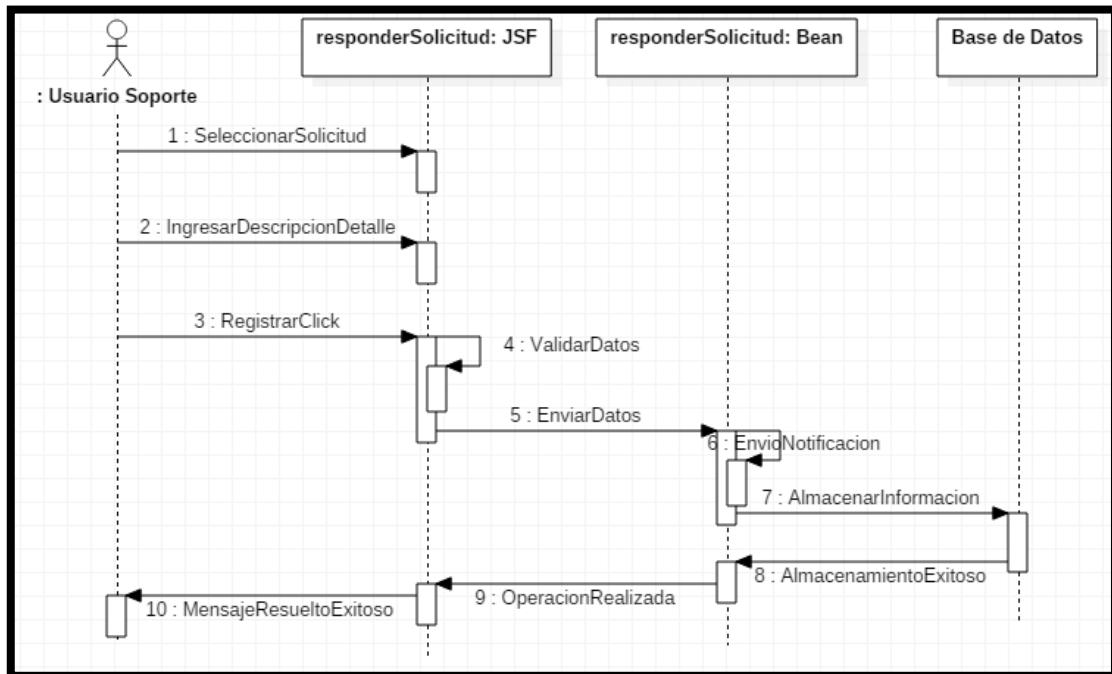
En la Figura 32 se muestra el diagrama de secuencia para poder visualizar las solicitudes que ha enviado un representante.



**Figura 32** Diagrama de Secuencia Histórico Solicitud

#### 4.6.5. Diagrama de Secuencia “Responder Solicitud”

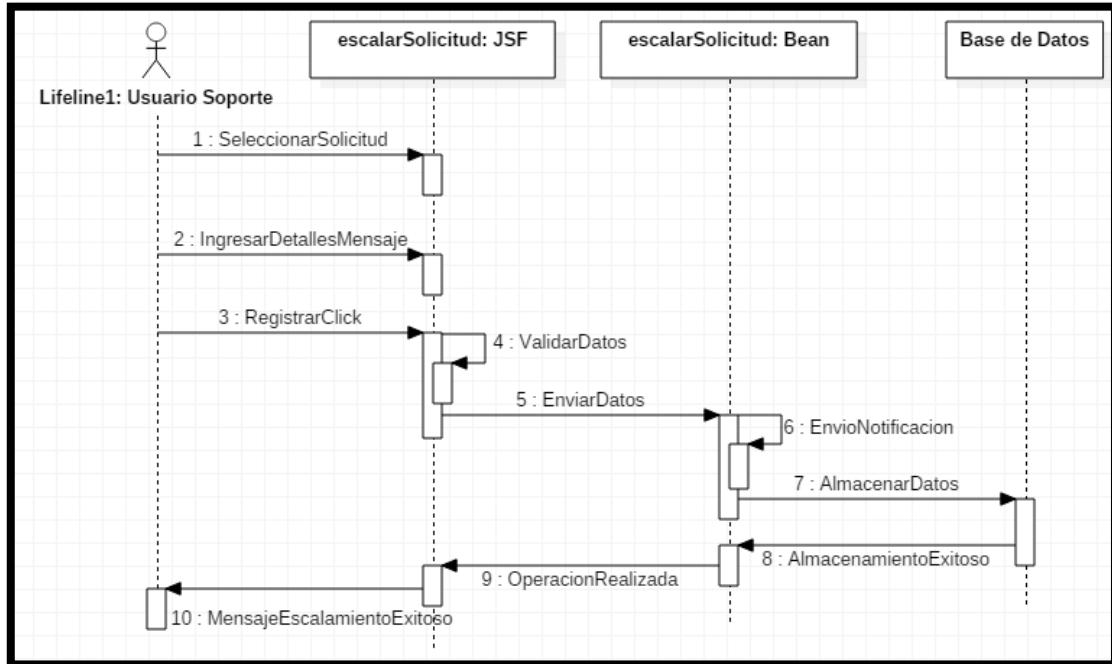
En la Figura 33 se muestra el diagrama de secuencia a realizar para dar respuesta a una solicitud recibida por el área de soporte.



**Figura 33** Diagrama de Secuencia Responder Solicitud

#### 4.6.6. Diagrama de Secuencia “Escalar Solicitud”

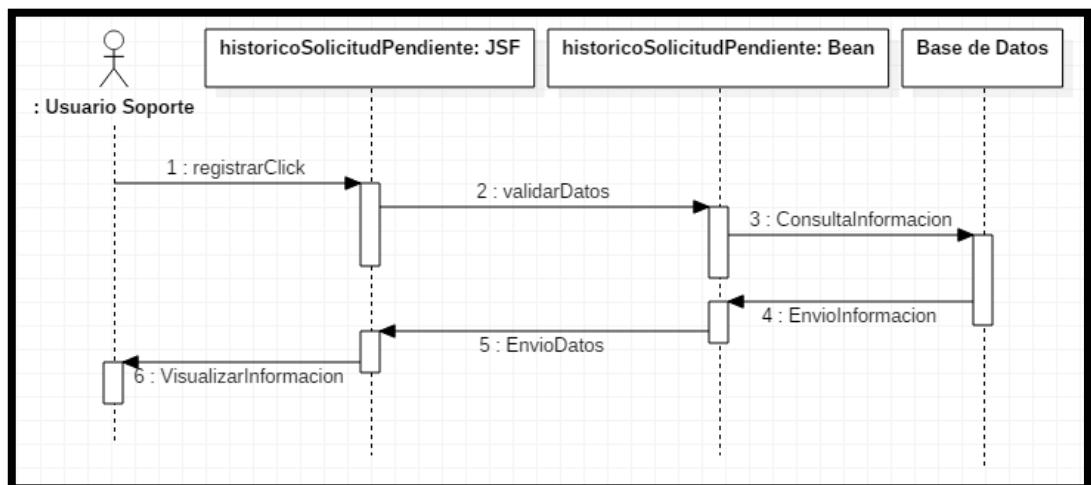
En la Figura 34 se muestra el diagrama de secuencia para realizar un escalamiento de una solicitud a otro usuario de soporte.



**Figura 34** Diagrama de Secuencia Escalar Solicitud

#### 4.6.7. Diagrama de Secuencia “Histórico Solicitud Pendiente”

En la Figura 35 se muestra el diagrama de secuencia para poder visualizar las solicitudes que se encuentran pendientes por resolver.

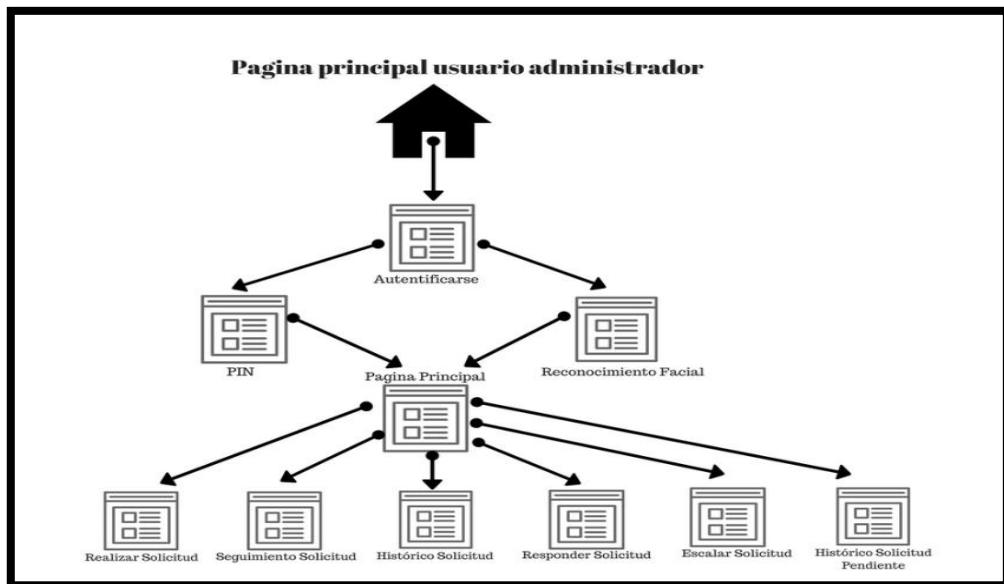


**Figura 35** Diagrama de Secuencia Histórico Solicitud Pendiente

## 4.7. Diagramas de Navegación

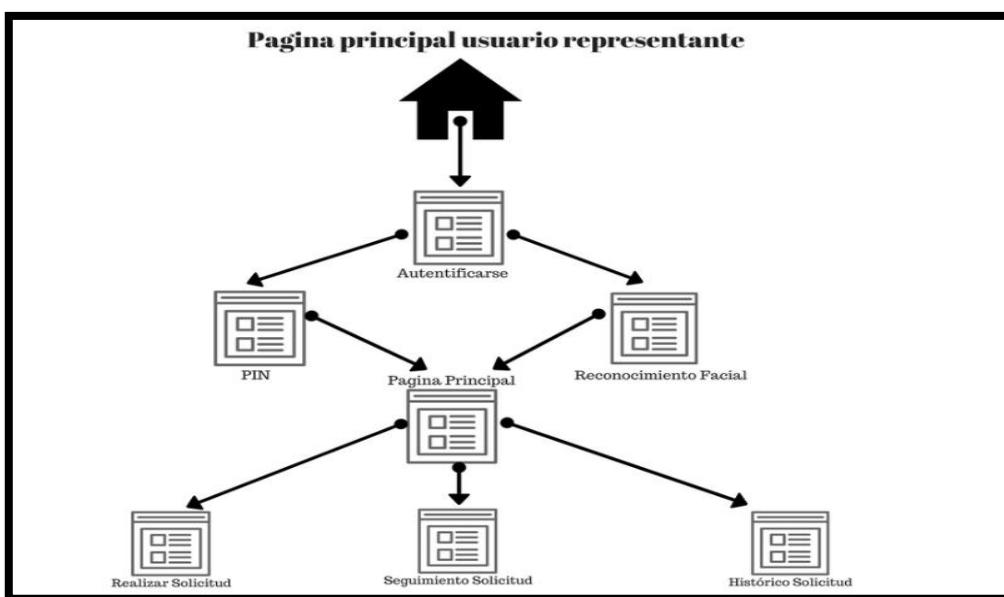
A continuación, se dará a conocer los diagramas de navegación de los tres usuarios pertenecientes al sistema AUTOMATZER.

### 4.7.1. Diagrama de Navegación “Usuario Administrador”



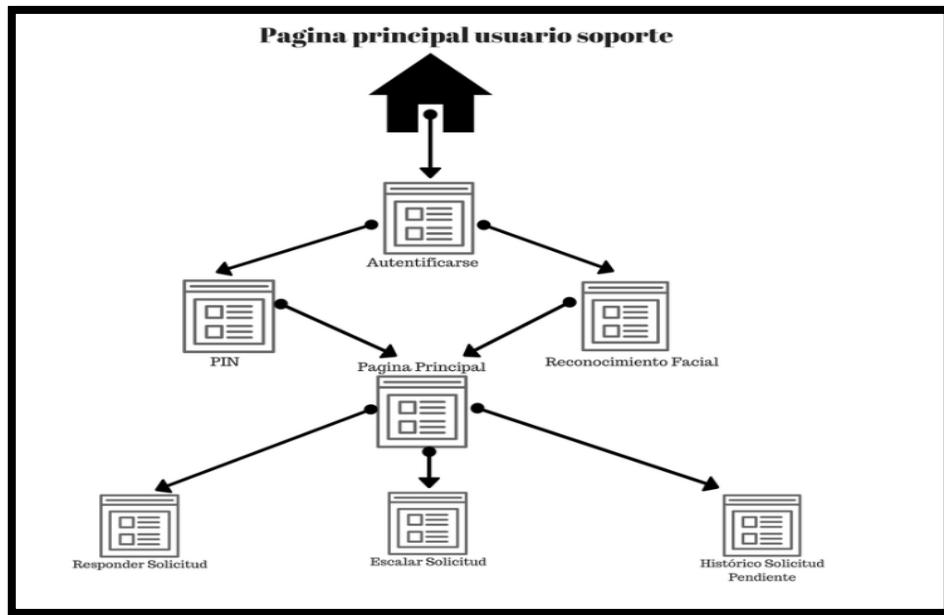
*Figura 36* Diagrama de Navegación Usuario Administrador

### 4.7.2. Diagrama de Navegación “Usuario Representante”



*Figura 37* Diagrama de Navegación Usuario Representante

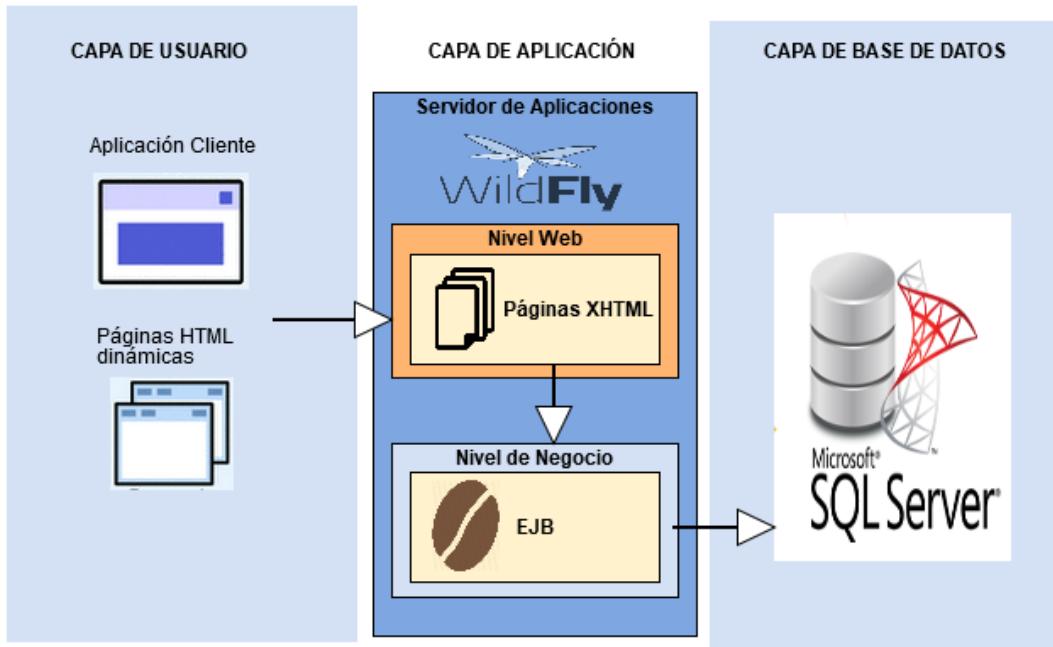
#### 4.7.3. Diagrama de Navegación “Usuario Soporte”



*Figura 38* Diagrama de Navegación Usuario Soporte

#### 4.8. Arquitectura del sistema

La arquitectura del sistema de soporte en línea AUTOMATZER es similar que la presentada en el capítulo anterior para el sistema de reconocimiento facial, con la diferencia de que contiene más componentes netamente del sistema de soporte.



*Figura 39* Arquitectura AUTOMATZER

El sistema AUTOMATZER se desarrolló en base a la especificación JEE, es decir como una aplicación empresarial. Siguiendo con la especificación JEE, se crearon los módulos WAR y EJB. JEE permite utilizar una arquitectura de N capas distribuidas dentro de los módulos anteriormente mencionados. Ambos módulos se ejecutarán sobre el servidor de aplicaciones Wildfly, que será el encargado de traducir las páginas XHTML a HTML para ser presentado por el navegador.

En el módulo EJB, en primer lugar, se mapearon las tablas de la base de datos utilizando Hibernate como motor de persistencia. El patrón DAO es implementado para el acceso a datos junto con JPA que nos permite interactuar con la base de datos con un enfoque orientado a objetos. Después de crear las clases dao, se procedió a desarrollar los servicios, los cuales serán consumidos dentro del módulo WAR.

Dentro del módulo WAR, se crean las páginas XHTML, que son las interfaces de usuario que se visualizarán en el navegador. Estas páginas hacen uso de beans de sesión llamados controladores, los mismos que utilizan los servicios desarrollados en el módulo EJB.

Como se aprecia en la Figura 39, la arquitectura del sistema AUTOMATZER tiene 3 capas. Iniciando por la capa de usuario, que por medio del navegador realiza solicitudes a la capa de aplicación. El servidor de aplicación contiene los niveles web y de negocio, representados por los módulos WAR, que recibe la solicitud, y EJB que posteriormente la procesa y se comunica con la capa de base de datos en caso de ser necesario.

En este capítulo se explicó el desarrollo e implementación de la aplicación AUTOMATZER, utilizando la metodología SCRUM, la cual nos ofreció muchas facilidades técnicas como es la optimización de tiempo, recursos, costos, y garantizo un desarrollo profesional para este proyecto y hacer las adaptaciones necesarias para optimizar los beneficios que el proyecto garantizará a Pfizer. Por otro lado, uno de los puntos clave que menciona SCRUM es el compromiso de todos los miembros del equipo, ya que esta metodología está enfocada en canalizar el esfuerzo de cada persona. Por lo tanto, el éxito de este proyecto de investigación recae en el compromiso y esfuerzo que ponen los miembros del equipo para lograr alcanzar las metas que se han planteado en este proyecto.

## CAPÍTULO V

### PRUEBAS Y ANÁLISIS DE RESULTADOS

En este capítulo se realizará las pruebas sobre el reconocimiento facial y se analizarán los resultados obtenidos. Además, se presentará el software AUTOMATZER a la empresa Pfizer con el fin de que se apruebe y den su conformidad con el mismo. Para realizar las pruebas del reconocimiento facial se presentará un escenario el cual se explicará a detalle a continuación.

#### 5.1. Escenario de prueba

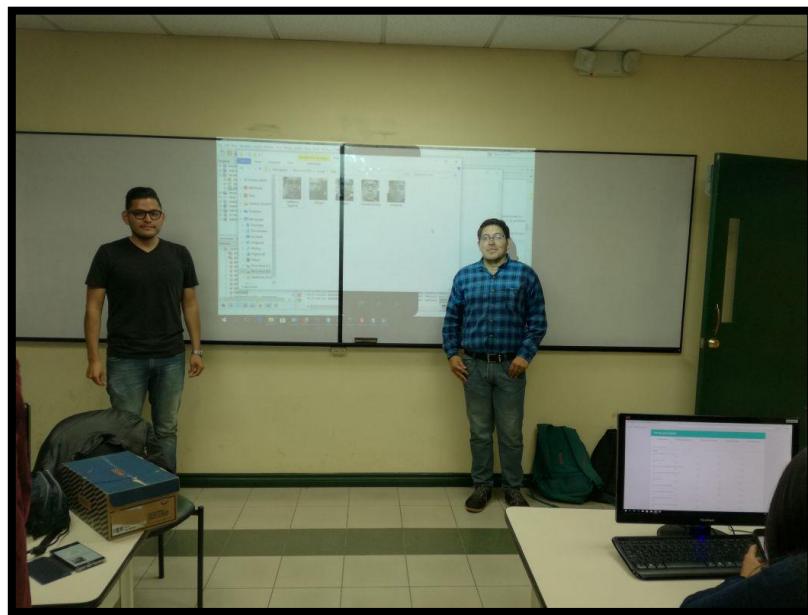
Las pruebas se realizaron en las aulas de la Universidad de las Fuerzas Armadas “ESPE”. Se escogieron 3 cursos al azar para que se realicen pruebas con el sistema de reconocimiento facial.

Dentro del algoritmo seguido para desarrollar el sistema de reconocimiento facial, se debe comparar la imagen procesada con las imágenes de la base de datos, y en función de un límite se decide si se reconoció o no el rostro. (Slavkovic & Jevtic, 2012) argumentan que este método requiere de una distancia mínima entre la imagen de prueba y las imágenes del conjunto de entrenamiento, ya que así la persona no se encuentre en la base de datos, esta podría ser reconocida. También dicen que no existe una fórmula para determinar este límite, ya que puede variar de acuerdo al conjunto de entrenamiento. Por este motivo, el propósito principal de estas pruebas es encontrar un límite adecuado en base a los reconocimientos exitosos que se realicen, además de realizar mejoras al sistema.

El escenario consta de 3 pasos que se describen a continuación:

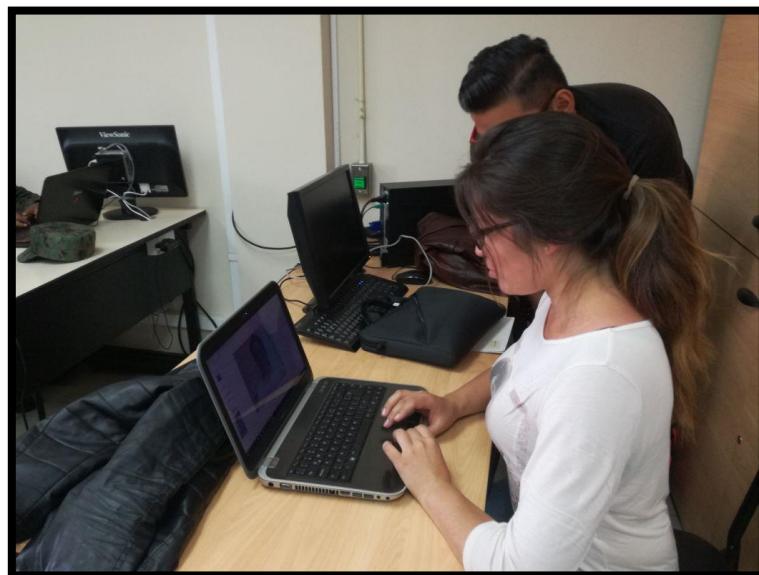
##### 5.1.1. Paso 1: Utilización del sistema de reconocimiento facial para entrenamiento

Para poder realizar esta prueba se dio una introducción de cuál era el objetivo de la misma. Se contó con la ayuda de 3 ingenieros de diferentes materias.



**Figura 40** Explicación Objetivo del Proyecto

Una vez iniciado el paso 1, el procedimiento que se llevó a cabo es que cada estudiante se tomó una fotografía con la cámara de la laptop que corría el sistema, para guardarla en la base de datos. El objetivo de esto, es realizar el entrenamiento previo para poder iniciar el reconocimiento de cada uno de los rostros de los alumnos que participan en la prueba.



**Figura 41** Fotografía de entrenamiento al sistema

### **5.1.2. Paso 2: Utilización del sistema de reconocimiento facial para reconocimiento**

Al terminar el paso 1, se contó con la información de los 45 sujetos de prueba en el conjunto de entrenamiento. Para proceder a realizar el reconocimiento facial se tomó una segunda fotografía a cada estudiante. El fin de este paso es conocer si el sistema logra reconocer los rostros de las nuevas fotografías y obtener la distancia mínima al rostro reconocido.

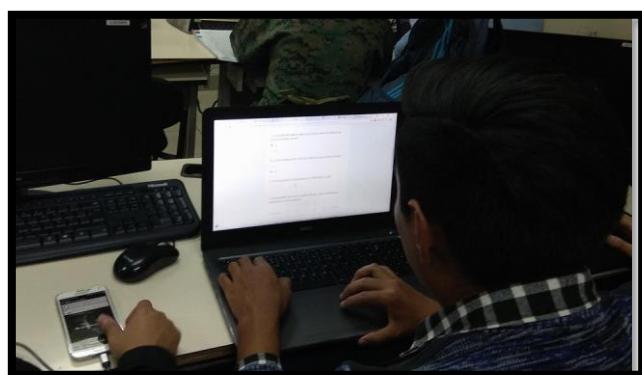


*Figura 42* Fotografía reconocimiento facial

### **5.1.3. Paso 3: Encuesta a los sujetos de prueba**

Finalizada la prueba, cada usuario debe llenar una encuesta, con el objetivo de saber su opinión de la aplicación y conocer si el sistema logró reconocer su rostro para poder comparar estos resultados con los que arrojó el sistema. De este modo, se podrán realizar las correcciones necesarias al sistema.

La encuesta se encontrará como Anexo D.



*Figura 43* Realización de la Encuesta

## 5.2. Tabulación de los resultados encuesta

La encuesta se realizó a 45 estudiantes, obteniendo los siguientes resultados:

**Tabla 23**

*Encuesta: Pregunta 1*

¿A qué grupo de edad pertenece?	Cantidad	Porcentaje (%)
<b>Menos de 20</b>	1	2
<b>20-29</b>	40	89
<b>30-39</b>	4	9
<b>Más de 40</b>		
<b>Total</b>	45	100



**Figura 44** Cuadro estadístico pregunta 1

De acuerdo a la Figura 44 de los 45 estudiantes encuestados el 89% están en edad entre 20 a 29 años.

**Tabla 24**

*Encuesta: Pregunta 2*

¿Cuál es su género?	Cantidad	Porcentaje (%)
<b>Femenino</b>	14	31
<b>Masculino</b>	31	69
<b>Total</b>	45	100



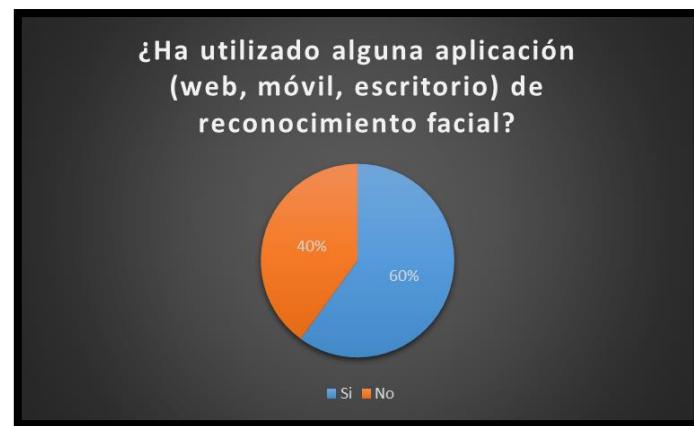
**Figura 45** Cuadro estadístico pregunta 2

De acuerdo a la Figura 45 se puede observar que la prueba se realizó más en hombres que en mujeres siendo el 69% de hombres y el 31% de mujeres

**Tabla 25**

*Encuesta: Pregunta 3*

¿Ha utilizado alguna aplicación (web, móvil, escritorio) de reconocimiento facial?	Cantidad	Porcentaje (%)
Si	27	60
No	18	40
Total	45	100



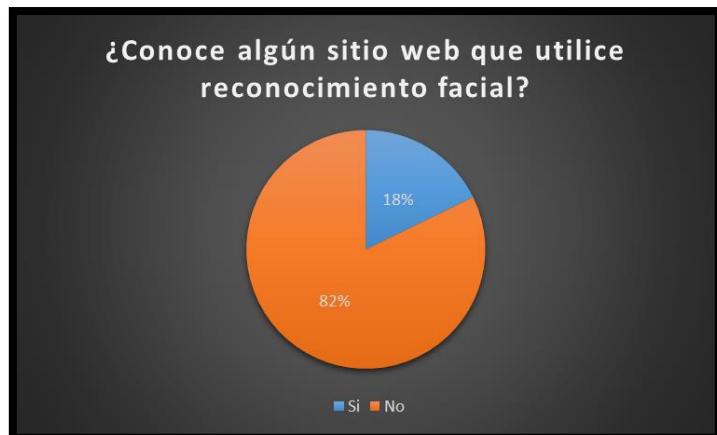
**Figura 46** Cuadro estadístico pregunta 3

De acuerdo con la Figura 46 el 60% de los estudiantes encuestados han utilizado una aplicación que utilice reconocimiento facial.

**Tabla 26**

*Encuesta: Pregunta 4*

¿Conoce algún sitio web que utilice reconocimiento facial?	Cantidad	Porcentaje (%)
Si	8	18
No	37	82
<b>Total</b>	<b>45</b>	<b>100</b>



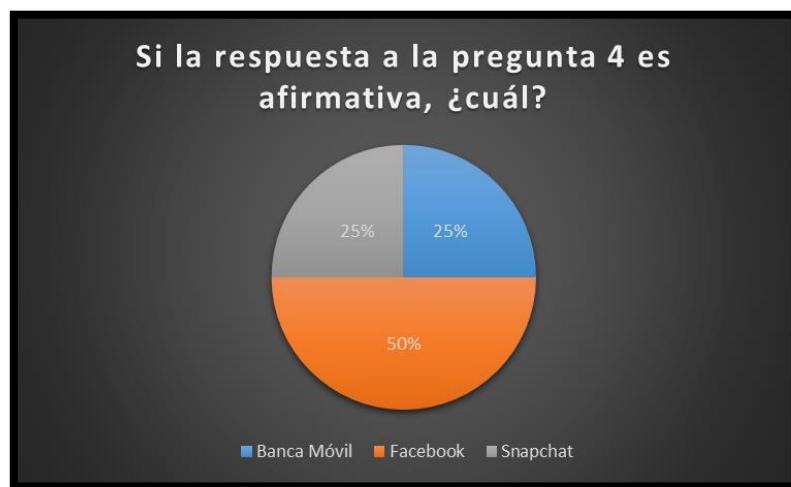
**Figura 47** Cuadro estadístico pregunta 4

De acuerdo a la Figura 47 solamente el 18% de los 45 estudiantes encuestados ha utilizado alguna vez una aplicación web que tenga reconocimiento facial.

**Tabla 27**

*Encuesta: Pregunta 5*

Si la respuesta a la pregunta 4 es afirmativa, ¿cuál?	Cantidad	Porcentaje (%)
Banca Móvil	2	25
Facebook	4	50
Snapchat	2	25
<b>Total</b>	<b>8</b>	<b>100</b>



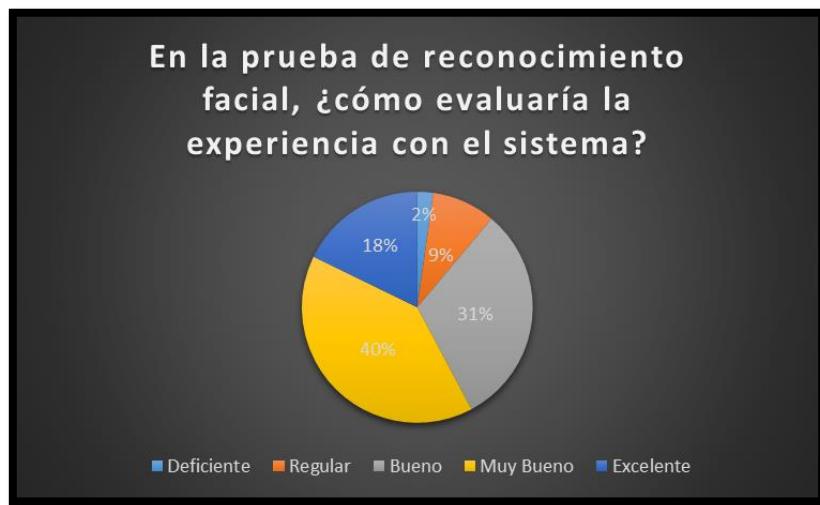
**Figura 48** Cuadro estadístico pregunta 5

De acuerdo con la Figura 48 del 18% que afirmaron en la pregunta 4 que han utilizado reconocimiento facial en una aplicación web, el 50% lo ha utilizado en Facebook.

**Tabla 28**

*Encuesta: Pregunta 6*

En la prueba de reconocimiento facial, Cantidad ¿cómo evaluaría la experiencia con el sistema?	Porcentaje (%)	
Deficiente	1	2
Regular	4	9
Bueno	14	31
Muy Bueno	18	40
Excelente	8	18
Total	45	100



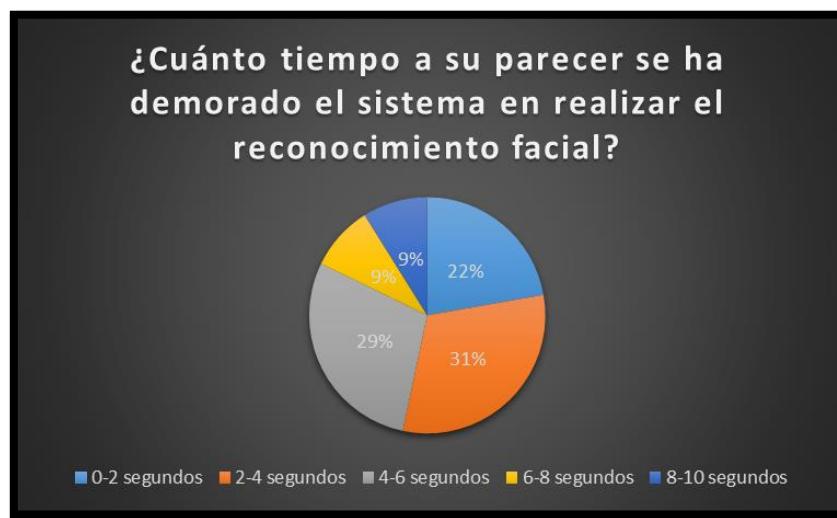
**Figura 49** Cuadro estadístico pregunta 6

De acuerdo a la Figura 49 de los 45 estudiantes encuestados el 40% obtuvo una muy buena experiencia con el sistema presentado, y tan solo el 2% ha tenido una experiencia deficiente.

**Tabla 29**

*Encuesta: Pregunta 7*

¿Cuánto tiempo a su parecer se ha demorado el sistema en realizar el reconocimiento facial?	Cantidad	Porcentaje (%)
0-2 segundos	10	22
2-4 segundos	14	31
4-6 segundos	13	29
6-8 segundos	4	9
8-10 segundos	4	9
Total	45	100



**Figura 50** Cuadro estadístico pregunta 7

De acuerdo con la Figura 50 de los 45 estudiantes encuestados el 31% afirmo que el sistema se demoró de 2 a 4 segundos en reconocer su rostro, el cual es un tiempo óptimo para el sistema.

**Tabla 30**

*Encuesta: Pregunta 8*

¿Al momento de realizar la prueba de reconocimiento facial el sistema logró reconocer su rostro?	Cantidad	Porcentaje (%)
Si	18	40
No	20	44
Otro rostro	7	16
Total	45	100



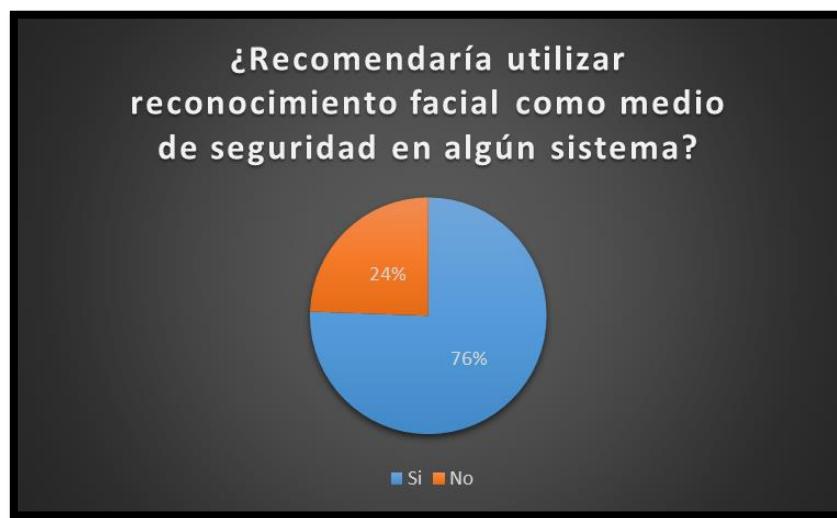
**Figura 51** Cuadro estadístico pregunta 8

De acuerdo con la Figura 51 de los 45 estudiantes que se realizaron la prueba de reconocimiento al 40% el sistema logró reconocer su rostro, el 16% reconoció el rostro de otra persona, y al 44% el sistema no logró reconocer su rostro.

**Tabla 31**

*Encuesta: Pregunta 9*

¿Recomendaría utilizar reconocimiento facial como medio de seguridad en algún sistema?	Cantidad	Porcentaje (%)
Si	34	76
No	11	24
Total	45	100



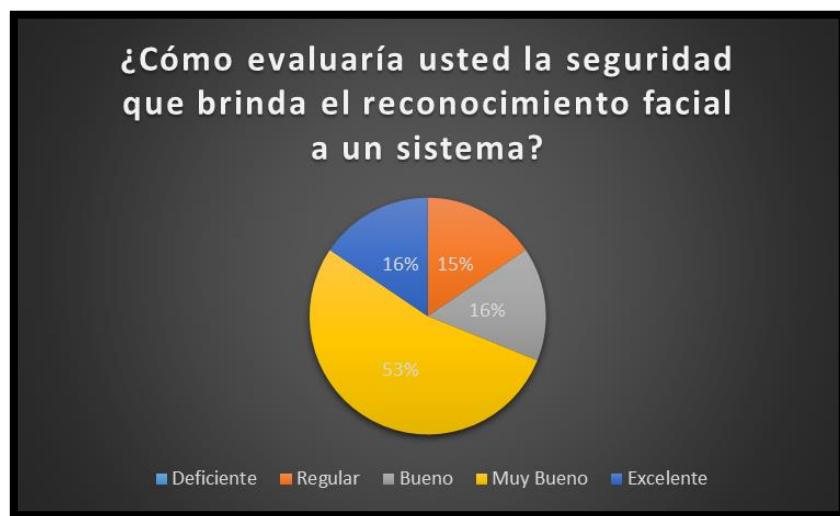
**Figura 52** Cuadro estadístico pregunta 9

De acuerdo a la Figura 52 de los 45 estudiantes encuestados el 76% recomendaría al reconocimiento facial como un medio de seguridad para un sistema

**Tabla 32**

*Encuesta: Pregunta 10*

¿Cómo evaluaría usted la seguridad que brinda el reconocimiento facial a un sistema?	Cantidad	Porcentaje (%)
<b>Deficiente</b>	0	0
<b>Regular</b>	7	15
<b>Bueno</b>	7	16
<b>Muy Bueno</b>	24	53
<b>Excelente</b>	7	16
<b>Total</b>	45	100



*Figura 53 Cuadro estadístico pregunta 10*

De acuerdo a la Figura 53 de los 45 estudiantes encuestados e 53% cree que la seguridad que brinda el reconocimiento facial es muy buena.

### **5.3. Análisis de Resultados**

El principal problema se notó en la pregunta 8, que se refiere al reconocimiento de rostro del alumno, donde el sistema no logró reconocer el rostro del 44% de los usuarios y el 16% reconoció el rostro de otra persona.

Según (Parmar & Mehta, 2013), el éxito del reconocimiento facial depende de varios factores como la luz, expresiones, iluminación, edad, transformaciones (traslado, rotación y escalamiento de imágenes), y poses. En la Figura 54 se observan nueve de las 45 fotografías de los sujetos de prueba. Como se puede notar, los rostros no están centrados de manera adecuada, y se distingue gran parte del fondo de la imagen.



**Figura 54** Rostros de prueba 1

De acuerdo a lo analizado en la prueba realizada, los resultados se dieron por las siguientes razones:

- **Iluminación:** En algunas fotos la iluminación fue mayor que en otras debido a la inclinación de la pantalla de la laptop al momento de hacer la fotografía. Como resultado, el sistema no logró reconocer todos los rostros o reconoció una fotografía de otra persona que tenía una iluminación parecida.
- **Resolución de la cámara:** Debido a que la prueba se llevó a cabo con la cámara de una laptop, la calidad de las fotografías no es la mejor. Esto afecta el proceso de reconocimiento ya que genera una diferencia más alta entre una foto y la otra por el valor de sus pixeles.
- **Fondo de la imagen:** Este problema se suscitó debido a que, al momento de tomar la fotografía en la fase de entrenamiento, estas fueron tomadas con un mismo fondo. Como se explica el procedimiento del algoritmo PCA en el [Capítulo III](#), cada pixel de la imagen representa un número, por lo tanto al momento de realizar el reconocimiento se toman en cuenta los pixeles del fondo de la imagen. Algunas fotos concordaban con el

fondo de otra, provocando que a varias personas las reconozca con otro rostro que le no pertenecía.

### 5.3.1. Solución a los problemas detectados

Luego de analizar los resultados obtenidos en el apartado anterior, se procedió a dar solución a los mismos, como se detalla a continuación:

- **Iluminación:** Con el fin de enmendar este problema, en el manual de usuario se recomendará que la fotografías para el reconocimiento facial sean tomadas en un lugar donde haya una adecuada iluminación.
- **Resolución de cámara:** El sistema se ejecutará en un Ipad que la empresa provee a sus trabajadores, las cuales poseen las mismas características, por lo tanto, todos los dispositivos contarán con una cámara de la misma resolución, tanto en la fase de entrenamiento como en la fase de reconocimiento facial. Otra medida que se tomó y que se detallará en el manual, es que la fotografía que se tome el usuario tendrá que ser con la cámara frontal del dispositivo.
- **Fondo de la Imagen:** Para subsanar este problema, se utilizó la librería OpenCV, que permite recortar la fotografía para que solo tome en cuenta el área del rostro. De este modo, al momento de almacenar las imágenes, solo se tomará en cuenta el rostro de la persona, independientemente del fondo.

Con estas mejoras al sistema se realizó una segunda prueba de reconocimiento facial, con ayuda de otros sujetos de prueba. Como se observa en la Figura 55, las fotografías son netamente del rostro de las personas, minimizando el fondo, lo cual optimiza el resultado del reconocimiento.



**Figura 55** Rostros de prueba 2

Los resultados obtenidos fueron los siguientes:

**Tabla 33**  
*Resultados segunda prueba*

<b>¿Al momento de realizar la prueba de reconocimiento facial el sistema logró reconocer su rostro?</b>	<b>Cantidad</b>	<b>Porcentaje (%)</b>
<b>Si</b>	41	91
<b>No</b>	4	9
<b>Otro rostro</b>	0	0
<b>Total</b>	45	100



**Figura 56** Cuadro estadístico de resultados segunda prueba

Como se puede observar en la Figura 56, el 91% de las veces el sistema de reconocimiento facial logró reconocer el rostro de los usuarios. Esto nos da a conocer que las mejoras realizadas en el sistema lograron remediar los problemas detectados, logrando así mejorar la eficacia del sistema al momento de realizar el reconocimiento facial.

#### **5.4. Carta de conformidad del proyecto AUTOMATZER.**

La empresa Pfizer realizó su evaluación de conformidad con el sistema AUTOMATZER. Luego de esto, el líder del equipo SMS proporcionó una carta de conformidad con el sistema, misma que estará como Anexo E.

Como se observó en este capítulo, se realizaron las pruebas y el análisis de resultados, con la finalidad de encontrar incidencias en el sistema de reconocimiento facial y así poder mejorarlo para que la empresa Pfizer pueda contar con un sistema óptimo acorde a sus necesidades, el cual es el principal objetivo de este trabajo de investigación. Además, como se mencionó al inicio del capítulo, en base a los reconocimientos exitosos, se pudo definir una límite adecuado para la comparación de los rostros, el cual permitirá integrar el reconocimiento facial al sistema de soporte en línea. Las mejoras que se hicieron en el sistema de reconocimiento facial permitieron definir la forma en la que el reconocimiento facial funcionara en el sistema AUTOMATZER.

## CAPÍTULO VI

### CONCLUSIONES Y RECOMENDACIONES

Para dar por terminado el presente proyecto de investigación, en este capítulo se expondrán las conclusiones y recomendaciones obtenidas a lo largo del mismo. Con esto se conocerá si se cumplieron los objetivos planteados al inicio y se dio la solución al problema planteado, que es la seguridad a la información manejada por los representantes de Pfizer.

#### **6.1. CONCLUSIONES**

El análisis de las diferentes técnicas de reconocimiento facial, permitió seleccionar la técnica adecuada para su implementación. Por medio de una comparación, se analizaron 4 características fundamentales que posee cada técnica y se le dio una ponderación a cada una. Con la implementación de la técnica seleccionada en el sistema AUTOMATZER se dio solución a la problemática que surge ante la seguridad de la información que manejan los representantes de Pfizer.

Para realizar la implementación de la técnica seleccionada en JAVA, se desarrolló un paquete con los métodos necesarios para el reconocimiento facial y otros métodos utilitarios para manipular imágenes, arreglos y valores numéricos. El desarrollo de este paquete fue de gran utilidad para la integración del reconocimiento facial en el sistema AUTOMATZER, ya que se simplificó el llamado a los métodos. También se puede concluir que, para un reconocimiento exitoso, las imágenes deben ser obtenidas bajo las condiciones similares de iluminación, calidad de imagen, etc.

Mediante la metodología SCRUM se logró realizar una planeación efectiva sobre el proyecto de investigación. Esta nos ofreció muchas facilidades como la optimización de tiempo, recursos, costos, y garantizó un desarrollo profesional para este proyecto.

Los resultados obtenidos al realizar las pruebas indican que el desarrollo del software AUTOMATZER es de utilidad para los representantes de Pfizer, señalando las pautas que se deben tomar al momento de utilizar el reconocimiento facial para ingresar al sistema.

#### **6.1. RECOMENDACIONES**

Para el desarrollo adecuado de un sistema empresarial se recomienda definir correctamente los procesos a seguir, manejar la documentación de requerimientos adecuada y

mantener una comunicación correcta con el cliente para lograr una satisfacción aceptable del sistema.

En el desarrollo de la metodología SCRUM se recomienda definir de forma clara los roles para optimizar los resultados que cada miembro del equipo obtiene mediante el cumplimiento de sus funciones.

El reconocimiento facial consiste en la aplicación de fórmulas matemáticas, que dependiendo del método que se utilice, pueden tener una alta complejidad. Se recomienda tener un conocimiento previo sobre el área matemática que se necesite para el entendimiento claro del proceso y la obtención de mejores resultados.

Para la correcta utilización del sistema de soporte en línea, se recomienda revisar detenidamente el manual de usuario. En este se explica el flujo de proceso que se sigue para realizar una solicitud, responderla y para la utilización del reconocimiento facial.

Debido a que el sistema corre sobre el servidor de aplicaciones Wildfly, es recomendable que este sea reiniciado cada día en una hora que no afecte a los usuarios, con el fin de limpiar cualquier tipo de fichero temporal que se almacene y mejorar su rendimiento.

## REFERENCIAS

- Beedle, M. (25 de Mayo de 2014). *SCRUM: An Extension Pattern Language for Hyperproductive Software Development*. Obtenido de [http://jeffsutherland.com/scrum/scrum\\_plop.pdf](http://jeffsutherland.com/scrum/scrum_plop.pdf).
- Chiang, K.-Y., Hsieh, C.-J., & Dhillon, I. (2016). Robust Principal Component Analysis with Side Information. *Proceedings of The 33rd International Conference on Machine Learning*.
- Cox, T. F., & Cox, M. A. (2001). *Multidimensional Scaling* (2nd ed.). London: Chapman & Hall/CRC.
- Davis West, J. (1 de Agosto de 2017). *Face First*. Obtenido de <https://www.facefirst.com/blog/brief-history-of-face-recognition-software/>
- Delbiaggio, N. (2017). *A comparison offacial recognition's algorithms*. Helsinki: Universidad de Ciencias Aplicadas Haaga-Helia .
- Eleyan, A., & Demirel, H. (2007). PCA and LDA based Neural Networks for Human Face Recognition. In *Face Recognition*. InTech., 16.
- Fallah, B., & Sodoudi, S. (2015). Bimodality and regime behavior in atmosphere – ocean interactions during the recent climate change. *Dynamics of Atmospheres and Oceans*, Vol. 70, 1-11.
- Fracastoro, G., Fosson, S. M., & Magli, E. (2017). Steerable Discrete Cosine Transform. *IEEE Transactions on Image Processing*, 26, 303-314.
- Gepshtain, S., & Keller, Y. (2015). Sensor Network Localization by Augmented Dual Embedding. Vol. 63, 2420-2431.
- Ghodsi, A. (2006). *Dimensionality Reduction A Short Tutorial*. Canada: Department of Statistics and Actuarial Science.
- He, X., & Niyogi, P. (2004). *Locality Preserving Projections*. Chicago: Advances in neural information processing systems.
- Hernández, R. G. (2010). *Estudio de técnicas de Reconocimiento Facial*. Barcelona.

- Jain, A., Flynn, P., & Ross, A. (2008). *Handbook of Biometrics*. New York: Springer.
- Jaiswal, S. (2011). Comparison between face recognition algorithm-eigenfaces, fisherfaces and elastic bunch graph matching. *Journal of Global Research in Computer Science*, 187-193.
- Kim, K. (2008). Face Recognition using Principal Component Analysis. *National Institute of Technology*.
- Lai, Z., Xu, Y., Chen, Q., Yang, J., & Zhang, D. (2014). Multilinear Sparse Principal Component Analysis. *IEEE transactions on neural networks and learning systems*, 1942-1950.
- Li, S. Z., & Jain, A. (2009). Face Recognition, Geometric vs. Appearance-Based. En L. Wolf, *Encyclopedia of Biometrics* (págs. 347-352). Springer US.
- Lu, X. (2003). Image Analysis for Face Recognition. *Department of Computer Science & Engineering, Michigan State University*.
- Parmar, D., & Mehta, B. (2013). Face Recognition Methods & Applications. *International Journal of Computer Technology and Applications*, 4, 84-86.
- Perlibakas, V. (2004). Distance Measures for PCA-based Face Recognition. *Pattern Recognition Letters*, 711-724.
- Saeed, K., & Nagashima, T. (2012). *Biometrics and Kansei Engineering*. Springer-Verlag New York.
- Schwaber, K., & Sutherland, J. (2013). *La Guía de Scrum*. The definitive guide to scrum: The rules of the game. Scrum.
- Sirovich, L., & Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, 519-524.
- Slavkovic, M., & Jevtic, D. (2012). Face Recognition Using Eigenfaces Approach. *Serbian Journal of Electrical Engineering*, 121-130.
- Tenenbaum, J. B., Silva, V. d., & Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, Vol. 290, 2319-2323.

- Turk, M., & Pentland, A. (1991). Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 71-86.
- Verma, R., Khurd, P., & Davatzikos, C. (2007). On Analyzing Diffusion Tensor Images by Identifying Manifold Structure Using Isomaps. *IEEE Transactions on Medical Imaging*, Vol. 26, 772-778.
- Xanthopoulos, P., Pardalos, P. M., & Trafalis, T. B. (2013). *Linear Discriminant Analysis*. Springer New York: Robust Data Mining.
- Yazdian, N., Tie, Y., & Venetsanopoulos, A. (2014). *Automatic Ontario license plate recognition using local normalization and intelligent character classification*. New York: IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE).
- Zang, B. (2011). Multiple features facial image retrieval by spectral regression and fuzzy aggregation approach. Vol. 4, 420-441.
- Zhao, W. (2003). Face Recognition: A Literature Survey. *ACM Computing Surveys*, 399-458 .