Implementación de un modelo de desarrollo evolutivo de software que permita detectar y mitigar ataques de ingeniería social utilizando técnicas de Deep Learning

Bosque Guanotasig, Santiago David y Zurita Bedoya, Bryan Abrahan

Departamento de Ciencias de la Computación

Carrera de Tecnologías de la Información

Trabajo de Integración Curricular, previo a la obtención del título de Ingeniero en Tecnologías de la Información

Ing. Fuertes Díaz, Walter Marcelo, PhD.

23 de agosto del 2023

Copyleaks

Plagiarism report

Tesis Zurita_Bosque 23ago23-Rev-WF....

Scan details

Scan time:	Total Pages:	Total Words:
August 25th, 2023 at 15:2 UTC	133	33157

Plagiarism Detection

Types of plagiarism		Words
Identical	3%	991
Minor Changes	0.3%	102
Paraphrased	2.2%	734
Omitted Words	0%	0

AI Content Detection

Text coverage
- AI text
- Human text

Plagiarism Results: (94)

Los 10 principios heurísticos de Jakob Nielsen | by P... 0.7%

https://medium.com/pildorasux/10-heuristicos-nielsen-abc9...

Píldoras UX

Open in appSign up Sign In Write Sign up Sign In Los 10 principios heurísticos de Jakob Nielsen Pí...

"They see me scrollin"—Lessons Learned from Inve... 0.5%

https://link.springer.com/chapter/10.1007/978-3-031-28643-...

Alia Saad

Skip to main c...

Reglas de Heurísticas de usabilidad | Marco Nuñez ... 0.5%

https://marco-nunez.com/docencia-mm/reglas-de-heuristica...

...

Firma

Firmado electrónicamente por:
WALTER MARCELO
FUERTES DIAZ

.....

Ing. Fuertes Díaz, Walter Marcelo, PhD

C. C. 1707017701

Departamento de Ciencias de la Computación

Carrera de Tecnologías de la Información

Certificación

Certifico que el trabajo de integración curricular: "**Implementación de un modelo de desarrollo evolutivo de software que permita detectar y mitigar ataques de ingeniería social utilizando técnicas de Deep Learning**" fue realizado por los señores **Bosque Guanotasig, Santiago David y Zurita Bedoya, Bryan Abrahan;** el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 20 de septiembre del 2023

Firmado electrónicamente por:
WALTER MARCELO FUERTES DIAZ

.....

Ing. Fuertes Díaz, Walter Marcelo, PhD

C. C. 1707017701

Departamento de Ciencias de la Computación

Carrera de Tecnologías de la Información

Responsabilidad de Autoría

Nosotros, **Bosque Guanotasig, Santiago David**, con cédula de ciudadanía N°1725653172 y **Zurita Bedoya, Bryan Abrahan**, con cédula de ciudadanía N° 1725264822, declaramos que el contenido, ideas y criterios del trabajo de integración curricular**: Título**: **Implementación de un modelo de desarrollo evolutivo de software que permita detectar y mitigar ataques de ingeniería social utilizando técnicas de Deep Learning** es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 20 de septiembre del 2023

…….

Bosque Guanotasig, Santiago David

C.C.: 1725653172

…….

Zurita Bedoya, Bryan Abrahan

C.C.: 1725264822

Departamento de Ciencias de la Computación

Carrera de Tecnologías de la Información

Autorización de Publicación

Nosotros **Bosque Guanotasig, Santiago David**, con cédula de ciudadanía N°1725653172 y **Zurita Bedoya, Bryan Abrahan**, con cédula de ciudadanía N° 1725264822, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de integración curricular: **Título: Implementación de un modelo de desarrollo evolutivo de software que permita detectar y mitigar ataques de ingeniería social utilizando técnicas de Deep Learning,** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Sangolquí, 20 de septiembre del 2023

.....	.....
Bosque Guanotasig, Santiago David	**Zurita Bedoya, Bryan Abrahan**
C.C.: 1725653172	C.C.: 1725264822

Dedicatoria

Este arduo trabajo lo dedico a Dios, que siempre ha sido mi fortaleza y la energía para poder culminar con éxito mi carrera.

A mi madre Marlene y mi padre Guillermo que han sido los autores principales de este gran logro con su esfuerzo y apoyo incondicional.

A mis hermanos, Paola y Juanfer que siempre estuvieron a mi lado y me ayudaron en todo momento.

A mis abuelitos, Rosita, Mamita y Papito que siempre estuvieron presentes y fueron como mis segundos padres.

Y de manera especial a mi abuelito Juanito que está en el cielo, que fue el principal motivo para que no me rindiera y me impulsará a dar lo mejor de mi cada día de mis estudios.

A mi familia en general y a todas las personas que fueron participes con una palabra, un gesto, un consejo hacia mi persona.

Bosque Guanotasig, Santiago David

Dedico este logro principalmente a Dios quien ha sido el ser todo poderoso el cual me ha ayudado y guiado a atravesar todas las dificultades que he atravesado a lo largo de mi vida. A su vez quiero dedicar este triunfo a mis padres Segundo Zurita y Carmita Bedoya los cuales han sido un pilar fundamental en mi formación personal inculcándome desde temprana edad todos los valores necesarios para ser un hombre de bien.

Además, deseo nombrar a mis queridos hermanos Silvia, Patricio y Freddy quienes de igual forma han representado un gran apoyo en todo momento, como si de un faro de luz en los momentos difíciles se tratase.

Por último, quiero agradecer a todos mis amigos y conocidos que contribuyeron significativamente en mi desarrollo personal y profesional llegando a ser una gran compañía en este viaje.

Zurita Bedoya, Bryan Abrahan

Agradecimientos

Agradecer es la mejor manera de ser reciproco con la vida.

A mi madre Marlene, que siempre estuvo con sus oraciones incansables y fue el gesto de amor más puro que me dio la vida. Sin ella no lo hubiera logrado.

A mi padre Guillermo, que me dio la firmeza y la virtud para realizar mis actividades de manera correcta, que me enseño el significado del esfuerzo y del trabajo constante para poder alcanzar mis metas y sueños.

A mi abuelito Juanito que siempre fue mi amor incondicional, mi padre, mi mejor amigo y mi ángel, que desde pequeño me hizo saber que nunca iba a estar solo y me dio la valentía de poder dar pasos firmes para poder tener la obtención de mi título de manera justa y transparente.

A mis compañeros que siempre estuvieron a mi lado con sus ocurrencias y alientos constantes, el compartir experiencias y momentos fue la mejor energía y el motivo para no bajar los brazos.

Al Dr. Walter Fuertes que encamino este gran proyecto y fue el eje principal para que se pueda culminar de manera exitosa.

Y a mi querida Universidad de las Fuerzas Armadas “ESPE” que fue la cuna que me acogió para poder crecer en conocimiento, valores, virtudes y habilidades que de manera segura me ayudarán en mis actividades profesionales.

Bosque Guanotasig, Santiago David

Agradezco a todo el equipo de docentes de todos los departamentos y áreas de conocimiento que conforman a la Universidad de las Fuerzas Armadas “ESPE”, en especial a todos y cada uno de aquellos docentes quienes me han ayudado a forjar profesionalmente con su paciencia, sabiduría y orientación.

Agradezco a todas nuestras autoridades del Departamento de Ciencias de la Computación las cuales a lo largo de mi formación académica me han guiado de la mejor forma. Gracias a nuestros tutores Ing. Fuertes Díaz, Walter Marcelo, PhD. y Ing. Mauricio Loachamín, Ph.D. por su tiempo, apoyo, enseñanzas y consejos los cuales fueron imprescindibles para conseguir esta meta.

Por último, gracias Freddy Zurita mi querido hermano, sé que tú me brindaste la fuerza necesaria en mis peores momentos y no me dejaste rendir, gracias a ti he llegado tan lejos para cumplir este sueño, uno de muchos más que deseo siempre estes presente guiándome y ayudándome a superar las dificultades que se presenten.

Zurita Bedoya, Bryan Abrahan

Índice de Contenido

Índice de Tablas

Índice de Figuras

Índice de Ecuaciones

Resumen

El Shoulder Surfing es un ataque de ingeniería social cuyo modus operandi se centra en sustraer información de la víctima mediante la “observación sobre su hombro”. Esto repercute en la seguridad de la información de la víctima, como de la organización a la que pertenece, de tal forma que puede causar daños financieros, reputacionales e incluso morales. El presente estudio busca detectar y mitigar esta problemática a través del desarrollo de un artefacto de software por medio de la implementación de tecnologías modernas como el Deep Learning a través del “Reconocimiento Facial” y las Redes Neuronales Convolucionales en conjunto con el Procesamiento de Imágenes y la Visión por Computador. Para lograrlo, se aplicó SCRUM por su peculiaridad para agilizar procesos, entrega de resultados y calidad en sus productos. La metodología para el módulo de reconocimiento facial se basó en un procedimiento de 9 pasos que van desde la carga y extracción de datos de las imágenes hasta la detección e identificación de personas, sin contar los pasos restantes de mitigación, registro y notificación adicionales implementados. Se realizaron pruebas de experiencia de usuario, rendimiento de la mitigación y rendimiento del reconocimiento facial. Los resultados muestran que el factor principal que determina las características de potencia para los dispositivos, es determinado por la resolución de video que es suministrado en tiempo real al sistema. Para el reconocimiento facial se desarrolló una ecuación para determinar el tiempo de procesamiento, que determinó que, por cada 5 imágenes, el tiempo incrementa 3.06 segundos aproximadamente de procesamiento. Estos resultados podrían incrementar el nivel de ciberseguridad en las personas más vulnerables de la familia, la academia, la empresa y la industria.

Palabras clave: Shoulder Surfing, Deep Learning, Face Recognition, Social Engineering, Mitigation.

Abstract

Shoulder Surfing is a social engineering attack whose modus operandi focuses on stealing information from the victim through "over-the-shoulder observation." This affects the security of the victim's information and the organization to which it belongs in such a way that it can cause financial, reputational, and even moral damage. The present study seeks to detect and mitigate this problem by developing a software artifact by implementing modern technologies such as Deep Learning through "Facial Recognition" and Convolutional Neural Networks in conjunction with Image Processing and Computer Vision. To achieve this, SCRUM was applied due to its peculiarity to streamline processes and deliver results and quality in its products. The methodology for the facial recognition module was based on a 9-step procedure that goes from loading and extracting data from images to detecting and identifying people, not counting the remaining steps of additional mitigation, registration, and notification implemented. Tests were performed on user experience, mitigation performance, and facial recognition performance. The results show that the main factor determining the devices' power characteristics is the video resolution supplied in real-time to the system. For facial recognition, an equation was developed to determine the processing time, which picked that for every five images, processing time increases by approximately 3.06 seconds. These results could improve cybersecurity in the most vulnerable people in the family, academia, business, and industry.

Keywords: Shoulder Surfing, Deep Learning, Face Recognition, Social Engineering, Mitigation.

Capítulo I: Introducción

Planteamiento del problema

Es normal observar cómo las personas se enfocan en la pantalla de sus dispositivos tecnológicos al ingresar, observar o enviar información, no obstante, la atención que prestan a todo el entorno a su alrededor es mínima o nula. Es por eso que surgen muchas vulnerabilidades de parte del factor humano que impiden que se pueda asegurar la información en su totalidad. Los seres humanos por naturaleza tienen el ímpetu de la intriga y tienden a observar cosas que no son suyas, ya sea por curiosidad o por querer obtener datos que puedan ayudar a concretar sus intereses.

La problemática surge cuando esos intereses tienden a ser con propósitos perversos, que pretenden perjudicar directamente al usuario vulnerado, atentando contra su información personal para poder aplicar diferentes acciones maliciosas como robos financieros, suplantación de identidad, violación de privacidad, extorsión y chantaje, entre otras. Por otra parte, en el ámbito empresarial el problema tiene una perspectiva más amplia, ya que cada empleado se convierte en un componente fundamental para el correcto funcionamiento de la misma, por ende, las empresas e instituciones requieren mecanismos para poder evitar que los propios empleados obtengan información de otros empleados y esto pueda causar inconvenientes en el correcto funcionamiento de producción dando apertura a que se perjudique su credibilidad en el mercado.

Cada persona puede adquirir conocimientos de cómo proteger su información para que esta no sea captada sin su consentimiento, esto se puede lograr mediante capacitaciones o autoaprendizaje, sin embargo, el hecho de que somos humanos es posible cometer errores y depositar confianza en individuos que puedan actuar de manera desleal o perjudicial. Es ahí donde la identificación y comprensión de los factores subyacentes en la seguridad de la

información se convierten en una tarea imperativa para abordar sus consecuencias. Además, se necesita plantear parámetros o herramientas que puedan contrarrestar estas aberturas, con el fin de mantener el ciclo de transmisión de la información sin ningún altercado, lo cual permite resguardar la confidencialidad, integridad y accesibilidad de la misma.

Justificación

Los ataques de Ingeniería Social son un tema de mucha relevancia en el ámbito de la ciberseguridad y en general de toda el área que conforma a las tecnologías de la información, esta problemática ha evolucionado con el propósito de explotar principalmente al eslabón más débil de la cadena de la información denominado “ser humano” por medio de la implementación de distintas metodologías, técnicas e incluso dispositivos que en conjunto persiguen el objetivo de que un sujeto A (atacante) obtenga beneficios de un sujeto B (víctima).

Existen varios tipos de ataques de Ingeniería Social de entre los cuales aquel que se busca idear una forma de mitigación en este trabajo de titulación es para el tipo de ataque “Shoulder Surfing”, el cual se basa en la sustracción de cualquier tipo de información personal importante de la víctima mientras se mira por encima de su hombro, aunque al parecer resulte sencillo de notar este tipo de ataque ha llegado a evolucionar conforme el paso de los años de tal manera que los cibercriminales han ideado una forma de perfeccionar la efectividad y la obtención de resultados óptimos por medio de la implementación de dispositivos de grabación que son posicionados estratégicamente de tal forma que no sean perceptibles a simple vista por la víctima, sin contar lo sofisticado y diminuto que puede llegar a ser este dispositivo, para esto es necesario un considerable esfuerzo e inversión por parte del atacante, sin embargo, es riesgoso por el posible rastreo e incautación de todos los dispositivos que haya utilizado.

Con base en lo anteriormente comentado, los atacantes suelen utilizar el método tradicional con el objetivo en mente de invertir la menor cantidad de recursos y no dejar rastro,

sin embargo, para que exista una gran diferencia en el éxito de este ataque se tiene que tomar en cuenta el entorno; es decir el ambiente de confianza existente entre todos los participantes de este ataque, ya que aunque resulte extremista, muchas de las veces este tipo de ataques son realizados por personas que tienen cierto grado de confianza, un claro ejemplo puede ser los compañeros de trabajo que por distintos motivos pueden llevar a cabo esto.

En vista de la complejidad de salvaguardar la información a todo momento mientras realizas a diario tus actividades en tu computador, se plantea la elaboración de una herramienta de seguridad en formato de aplicación de escritorio que parte del concepto de dispositivo personal, el cual se comprende como un dispositivo que es para uso único y exclusivo de una persona, de tal manera que todo lo que realice el usuario en su propio dispositivo no deberá estar sujeto a sustracción de información de ningún tipo

Objetivo General

Diseñar, desarrollar e implantar un modelo de desarrollo evolutivo de software que permita detectar y mitigar ataques de Ingeniería Social utilizando técnicas de Deep Learning con el fin de obtener mecanismos de discriminación del contenido malicioso que incrementa los niveles de inseguridad de la información.

Objetivos Específicos

Realizar la fase de análisis que incluya el estado del arte y el marco teórico referencial con el fin de fundamentar la problemática y posibles soluciones tecnológicas.

Determinar la precisión de los algoritmos de Deep Learning e iniciar la fase de diseño de la arquitectura del modelo de desarrollo evolutivo con el fin de obtener una descripción detallada del software que incluye programas, algoritmos reutilizables, diseño de la base de datos, diseño de interfaces, usabilidad y navegabilidad.

Realizar la fase del Diseño e implementación del modelo de desarrollo evolutivo de software que permita detectar ataques de Ingeniería Social utilizando Deep Learning.

Realizar pruebas e implantación del modelo de desarrollo evolutivo de software y elaborar la documentación necesaria según el progreso y las acciones que se realicen.

Evaluar e interpretar los resultados de las pruebas de validación e implantación del artefacto de software, además de elaborar los manuales técnicos y de usuario.

Alcance

Para el desarrollo del presente tema de trabajo de titulación se propone diseñar y desarrollar una aplicación para la plataforma de escritorio, sin dejar de lado la posibilidad de escalar a una aplicación multiplataforma para entornos web o móvil, con el fin de satisfacer las exigencias, necesidades y requerimientos cambiantes de las arquitecturas de software de las organizaciones.

Esta aplicación tiene por objetivo detectar y mitigar ataques de Shoulder Surfing principalmente para dispositivos personales con sistema operativo Windows 10 que dispongan de una video cámara. La detección y mitigación se pretende realizar mediante la implementación de una potente herramienta de reconocimiento facial la cual utiliza algoritmos de Deep Learning, en concreto el algoritmo que corresponde a un modelo de red neuronal perfecta para el procesamiento de las imágenes como lo es el modelo de Red Neuronal Convolucional o más conocida por sus siglas en inglés CNN, la cual aportará con la detección y reconocimiento/identificación de rostros. Esta herramienta se la utilizará y repotenciará de tal manera que adquiera la característica evolutiva de añadir nuevos miembros de una organización.

Finalmente, se utilizará tecnologías de organización, recopilación y estructuración de datos, tales como base de datos SQL, para que la aplicación pueda gestionar automáticamente la información de los dispositivos personales, usuarios y recopilación de ataques que ha sufrido el usuario con respecto de su dispositivo y el atacante, solo cuando la arquitectura de software diseñada haya sido alimentada con la información necesaria.

Resultados esperados

El propósito de este trabajo es poder crear un mecanismo que pueda contrarrestar los ataques de Shoulder Surfing, con el fin de poder proteger la información de empresas, instituciones y usuarios que puedan estar expuestos a estos altercados.

Con lo previamente descrito, se busca crear una aplicación de escritorio para poder detectar dichos ataques, esto enfocada en un ambiente empresarial o educativo, que logre captar al propietario del equipo para hacer una relación de pertenencia mediante reconocimiento facial que estén vinculados con la información del computador y así poder solventar el problema de robo de información, en el caso de que terceros quieran acceder sin consentimiento al equipo. Dicho esto, la aplicación tendrá acciones de mitigación y notificación con el objetivo de que el ataque no se concrete y sea notificado oportunamente al usuario de la posible vulneración de su información. Además, de manera visual se busca desarrollar una sección que describa estadísticas de los posibles ataques mediante un balance de horarios y actividades. Con esto se puede lograr que empresas e instituciones puedan resguardar la información y puedan percatarse del porcentaje de incidentes que posee su organización.

Capítulo II: Fundamentación teórica y estado del arte

Fundamentación teórica

Poder comprender la importancia de la seguridad de la información y de cómo tener un sistema que pueda gestionar la misma, es el primer paso para empezar en la ardua labor de resguardar la información, además, entender que la información se puede quebrantar tanto de manera lógica como física. Esto permite adquirir conocimientos de factores de riesgo y prevención, seguridad física y gestión acceso, así mismo el poder hacerles frente a incidentes de ataques externos e internos, conlleva el apoyo de varias tecnologías como la inteligencia artificial, la detección y reconocimiento facial, Machine Learning y Deep Learning, esto con el único propósito de contrarrestar los ataques mencionados, específicamente los de Ingeniería Social que son del tipo Shoulder Surfing. Es por eso que es indispensable conocer cómo funciona, ¿cuáles son los escenarios en que se puede presentar? y ¿cuáles han sido las herramientas que se han elaborado para detectar y mitigar dicho ataque?

Seguridad de la Información

La Seguridad de la Información es muy importante para poder proteger los datos sensibles y garantizar la integridad, confidencialidad y disponibilidad de la misma. En la actualidad las empresas tienen su productividad adherida a la tecnología, y confían en que esa relación pueda garantizarles un distintivo posicionamiento jerárquico sobre la competencia y pueda brindar seguridad y confianza a sus empleados y clientes. En instituciones bancarias, de salud, educación, telecomunicaciones, aseguradoras, entidades públicas, entre otras la información que se maneja se convierte en el factor crítico de su existencia, ya que mediante los datos que manejan pueden brindar su servicio u organizar sus actividades. Las amenazas a la información se pueden categorizar en desastres ambientales, errores y accidentes humanos, averías en las tecnologías y sistemas de información, y por intenciones maliciosas de agentes internos o externos. Estos escenarios mencionados pueden desembocar en un sin número de

problemas como la utilización de virus y código contaminado, suplantación de identidad, espionaje, extorsión y manipulación, uso no autorizado de base de datos, entre otros (Figueroa-Suárez et al., 2018).

Por todo lo antes mencionado, las organizaciones se han enfocado en implementar medidas y políticas para prevenir, detectar y responder las amenazas y vulnerabilidades que puedan comprometer la seguridad de los activos informáticos. Esto incluye el uso de métodos de protección como la criptografía, la gestión de identidad y acceso, y la seguridad en redes y comunicaciones. Además, se promueve la conciencia y capacitación del factor humano, ya que las personas también son un componente fundamental en la seguridad de la información, considerado el punto más vulnerable (JAVIER, 2008).

SGSI (Sistema de Gestión de Seguridad de la Información)

El SGSI se centra en una consolidación de normas y directrices, que toman en cuenta los procedimientos y los recursos que una organización posee para administrar, gestionar y garantizar la seguridad de la información, esto con el fin de proteger la información crítica de la institución y siga su correcto accionar de manera óptima y segura (*SGSI*, s. f.).

Cabe recalcar que para poder implementar el SGSI de una manera correcta se apoya del estándar internacional ISO/IEC 27001, que brinda una guía para poder realizar un sistema de gestión de seguridad de la información, esto para organizaciones pequeñas, medianas y grandes, en todos los ámbitos y sectores empresariales ya que provee información de todo tipo que se acopla a cada organización según las necesidades y los requerimientos que esta tenga (14:00-17:00, s. f.).

Para poder definir el alcance que tendrá el SGSI, la empresa que lo implemente debe hacer una revisión previa de las áreas y de los activos que son el motor para que esta tenga funcionamiento, es decir los puntos críticos, que con el robo, ausencia o denegación de estos

servicios la empresa tiende a perder productividad y estar en un estado de riesgo, esto se evalúa de una manera previa al simular escenarios que pueden ocurrir para determinar medidas que se deberían tomar para poder controlar o erradicar estos sucesos. Hay que tomar en cuenta que estos escenarios se producen gracias a las vulnerabilidades que tiene la empresa, estos atentados pueden ser amenazas involuntarias que corresponden a desastres naturales y fallas de personal por descuidos o falta de conocimiento, por otro lado, existen las amenazas voluntarias en donde vienen inmersas los ataques externos e internos.

Los ataques externos se pueden producir por personas ajenas a la empresa, que pertenecen a la competencia o que simplemente se dedican a perturbar y alterar la información ajena con el objetivo de buscar recompensas monetarias, a estas personas también se les conoce como terroristas o cibercriminales. Mientras que los ataques internos son los mismos empleados que por motivos personales o desacuerdos laborales (peleas, despidos, descontentos) buscan causar daño a la empresa, por otro lado, pueden ser personas que trabajaron allí y su conocimiento de credenciales o de información sensible es mal utilizada (Romero Castro et al., 2018). En este sentido los empleados que laboran y pretenden robar información usan varios métodos para lograrlo, el constante monitoreo del atacado para poder recolectar información privada como contraseñas, documentos, textos y material audiovisual es el principal objetivo del atacante, ahí es donde se utiliza la mal denominada astucia humana o conocida como Ingeniería Social, que mediante sabotajes, facilidad de palabra o husmear sin consentimiento se vuelve un mecanismo muy poderoso que no es difícil de aplicar a la hora de obtener datos. Esto también es mencionado en (Benavides-Astudillo et al., 2022) donde se indica a la Ingeniería Social como uno de los ataques más efectivos de ciberseguridad, en el cual el atacante busca engañar al usuario final con el objetivo de ejecutar actividades delictivas cibernéticas.

Justamente por esto es que el SGSI se encarga de tapar y contrarrestar los posibles daños futuros para poder evitar pérdidas de magnitudes considerables, ya sea de manera física o lógica. Se necesitan armas para combatir las malas intenciones que puedan atentar contra su prestigio y su seguridad, de allí que se han creado varios mecanismos que ayudan a las empresas a fortalecerse contra la delincuencia cotidiana e informática.

Seguridad Física y Control de Acceso

La seguridad física y el control de acceso son puntos esenciales que deben estar presentes en una organización si se quiere garantizar la protección de información y de sus recursos. Como su palabra mismo lo indica se refiere a la parte tangible de la organización, como poder cuidar y mitigar las amenazas que se pueden dar en estos espacios. El punto de partida es hacer una auditoría de toda la empresa en la parte infraestructural, para verificar todos los puntos de acceso que esta tenga, principalmente por las puertas de entrada, comprobar en qué condiciones se encuentran las chapas, candados, cerraduras y la estructura de las puertas, en caso que no necesiten un cambio inmediato hacer una proyección de vida de uso de cada objeto para poder determinar la adquisición de nuevos elementos. Posteriormente se proseguiría por las cercas de seguridad con el propósito de verificar que funcionen correctamente y así solventar agujeros en donde no se cumpla su trabajo en su totalidad. Las ventanas y paredes de vidrio se deben fortalecer mediante rejas de metal o sensores que puedan alertar si son vulneradas. No olvidar a los parqueaderos que es el punto en donde hay el ingreso y la salida de la organización, que no se pueden revisar de manera exhaustiva por el tamaño de los vehículos, y pueden ser herramientas para el ingreso de armas, estupefacientes, explosivos, entre otros, que puedan usarse para atacar a la organización. Todo lo mencionado anteriormente puede tener un aliado sumamente estratégico, que es la implementación de cámaras de vigilancia, estos artefactos pueden potenciar las puertas de entrada para verificar la identidad de los empleados, además pueden dar una perspectiva más clara de lo que sucede

dentro y fuera de la empresa para asimilar los riesgos a los que se podría enfrentar (*Physical Access Control - an overview | ScienceDirect Topics*, s. f.).

Las normas de una organización son el filtro para poder establecer métricas de acceso, una empresa debe regirse a un reglamento para tener un correcto ingreso y salida de sus empleados. El control de acceso se basa en otorgar permisos y roles según las actividades que realicen, esto con el fin de tener rangos de accesibilidad y de manejo administrativo para un discernimiento y distribución de los lugares que los empleados puedan acceder de manera física o tecnológica. El control de acceso es muy importante ya que refuerza y potencia la seguridad física, un claro ejemplo son las tarjetas de acceso con el cual es más difícil que una persona extraña a la empresa logre obtenerla. La autenticación biométrica es una de las herramientas más confiables para tener un correcto acceso, ya que involucra las partes del cuerpo del propietario como el iris de los ojos, las huellas dactilares y el rostro como tal, que mediante algoritmos de detección facial hacen que atravesar estas barreras sea un trabajo más difícil para el delincuente. Por otro lado, las contraseñas, pines y patrones de seguridad son mecanismos que ayudan a controlar el acceso del usuario, ya que es información que solo el propietario conoce, para reforzar estos mecanismos, lo que se hace es combinar la autenticación de dos hasta tres factores, puede surgir contraseñas acompañadas de detección biométrica y también patrones o pines acompañadas de tarjeta de acceso. Todo esto es referente a la parte del empleado, sin embargo, suelen haber grietas de información cuando tenemos la necesidad de conectarnos a internet, ahí los patrones de acceso se vuelven más complejos ya que la abertura que tenemos en la red es muy vasta y peligrosa, por lo que se opta por segmentar la red con ayuda de firewalls y VLANS, para poder hacer que si surge un ataque no afecte a toda la organización sino solo a una parte, lo que facilita la corrección de una manera más rápida y conservada (*Physical Security Control - an overview | ScienceDirect Topics*, s. f.)

Por último, es importante recalcar que todos estos mecanismos de protección ayudan a tener un resguardo óptimo, sin embargo, el punto débil de toda organización será el factor humano, por lo que es necesario tener una constante capacitación a los empleados para enseñarles cómo evitar y disminuir estos riesgos, haciéndoles partícipes de simulacros, charlas de seguridad informática, y presentándoles las consecuencias que habría si ellos no actúan de una manera óptima. El hecho de tener normas de acceso y de seguridad física no garantiza una protección en su totalidad si los protagonistas de estas acciones no lo realizan de una manera correcta (*Physical Access Control - an overview* | *ScienceDirect Topics*, s. f.)

Gestión de Identidad y Acceso

La gestión de identidad y acceso o por sus siglas en inglés IAM (Identity and Access Management), se centra principalmente en brindar una ayuda a las empresas, para que los empleados y artefactos puedan obtener los permisos correspondientes y ocupen lo que necesiten según sus actividades. Esto suena algo muy básico no obstante para las empresas poder automatizar los procesos de credenciales (username, contraseña) de todos sus empleados y que según su rol puedan obtener un acceso inmediato a los recursos que necesitan, es algo que beneficia de manera excepcional a la organización, lo cual permite dejar a un lado el proceso tradicional que lo hiciera un empleado quien debería revisar la función que iba hacer su compañero y luego darle permisos diferentes según la actividad que realizaba, esto era algo tedioso ya que debe hacerlo uno por uno, al igual que si un programa necesita acceso a un determinado puerto, revisar sus características y porque necesita ese acceso, si un periférico necesita acceder a la información multimedia, determinar de qué fecha a que fecha debe tener esa información, era algo sumamente largo. A medida que la tecnología avanza y se hace presente el IoT, más artefactos necesitan permisos para poder realizar su tarea de manera óptima, esto se vuelve algo peligroso ya que un mal acceso podría perjudicar a la empresa o detener su producción. Es por eso que la IAM ayuda a gestionar todo lo que se

refiere a credenciales del personal, clientes y proveedores que necesiten consumir sus servicios. El IAM se encarga de proteger, gestionar y administrar toda la información crítica, solventa problemas con bloqueos o accesos inmediatos y protege de ataques cibernéticos que puedan apuntar al robo de información. Esto trae beneficios a la empresa ya que puede estar tranquilo de que las actividades productivas puedan seguir su curso, sin detenerse por sucesos inminentes como el ingreso o salida de empleados, sentirse seguro de que si alguien fue contratado tiene todo lo que necesita a su disposición según el cargo que va ejercer, y si alguien tuvo que salir de la organización, no pueda tratar de acceder o de manejar la información que antes le pertenecía como trabajador. Del mismo modo si un cliente tiene acceso a información de la empresa como tal y si deja de utilizar los servicios, de inmediato se quite todo el acceso que tuvo, también si un proveedor pierde el vínculo con la organización, levantar sus accesos y asegurar la información que se obtuvo hasta el último día de servicio. Todo esto hace que la eficiencia incremente en la empresa, que se use lo necesario, en el momento indicado, con el permiso oportuno y el tiempo justo.

Tipos de Autenticación

Es algo espectacular todo lo que brinda el IAM, no obstante, es algo que se necesita conocer. La parte de autenticación se realiza de manera tradicional, se necesita de un usuario y contraseña, y luego de aquello surge los permisos que tiene dicho usuario según las actividades que va a desarrollar. Existen varios tipos de autenticación; el inicio de sesión único o por sus siglas en inglés SSO (Single Sign-On), este tipo se encarga de guardar tus credenciales para abrir varias aplicaciones sin necesidad de ingresar nuevamente el usuario y contraseña, esto beneficia al usuario en ahorro de tiempo y aumenta la versatilidad para trabajar en distintas aplicaciones, la desventaja puede ser que al usar solo una vez la información, esta puede ser olvidada, por otra parte beneficia en que terceros no puedan robar tu información, ya que está previamente guardada y verificada, para así no realizar ningún

proceso de acceso. Tenemos la autenticación multifactor que potencia la seguridad de ingreso, ya que además de digitar el usuario y contraseña, aumenta el ingreso de un código que puede ser enviado por mensajes de textos, correo o incluso por llamada, este código tiene un límite de tiempo y solo se puede usar una vez. Para terminar, tenemos la autenticación basada en riesgos, esto implementa la autenticación multifactor, sin embargo, la diferencia es que se activa cuando el ingreso se realiza de manera sospechosa o desde una ubicación no habitual, es decir desde otro país o fuera del área de trabajo que se encuentre normalmente, además detecta malware por lo que presenta una barrera para que el usuario se alerte y pueda verificar el correcto acceso con normalidad (*IBM Security Verify - Autenticación avanzada*, 2022).

Factores de riesgo y prevención

En al ámbito digital se presentan factores de riesgos, todo estos con el fin de producir afectaciones al individuo u organización, en este apartado directamente a la información y a la privacidad. Estos factores de riesgos pueden presentarse desde varios ámbitos como riesgos personales que afectan directamente al acoso y al ataque de información delicada, esto deriva en riesgos económicos, ya que los malhechores se aprovechan de su ventaja y actúan mediante extorsiones, así se aprovechan del limitado conocimiento del usuario en este caso de las personas adultas que tienden a sufrir estafas por la suplantación de sitios web o por correo maliciosos. De igual manera los mismos propietarios pueden ser un riesgo por desconocimiento y escaso cuidado con su información, ya que algunos actúan de una manera tranquila y dejan a disposición su información sin darse cuenta de las consecuencias que puede pasar a futuro. Otro riesgo que está presente, viene involucrado directamente en el artefacto o software, ya que aplicaciones, dispositivos y programas pueden tener fugas de información, falta de cifrado y puertos abiertos sin consentimiento, lo que provoca que estén vulnerables ante los delincuentes (Rojas-Díaz & Yepes-Londoño, 2022).

Ante toda esta problemática se han creado métodos de prevención que ayudan a disminuir el riesgo e incrementar la seguridad, para tener un punto de partida se puede empezar por la correcta verificación de identificación y autentificación, hay mecanismos que ya brindan ese servicio como el IAM que también gestiona y autoriza los controles para el acceso a roles determinados. En la parte de internet se han creado diseños de redes para la correcta segmentación y así poder implementar firewalls, filtrar paquetes, inspeccionar en qué estado se encuentran estos paquetes para que puedan dar paso a su recorrido. Todo esto desemboca en la protección de tráfico de red, para poder interceptar los datos, consolidar la exposición inalámbrica y fortalecer el uso de redes privadas virtuales (VPN) (Vega Briceño, 2021).

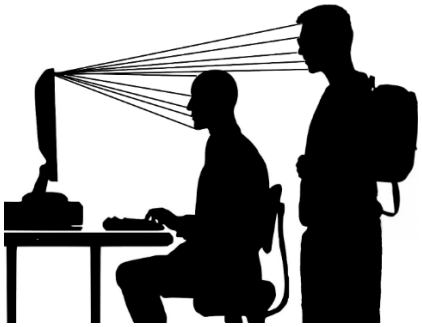
Así como existen miles de riesgos hay miles de prevenciones, el que se logre evitar ataques y afectaciones, se centra en el usuario o persona encargada de gestionar e implementar bien los métodos que ayudan a solventar problemas, seguir y ayudarse de métricas y guías proporciona una óptima seguridad al igual que tener buenas prácticas como factor humano, potencia y refuerza las ayudas tecnológicas.

Shoulder Surfing

El Shoulder Surfing para (Bošnjak & Brumen, 2020), (Behera et al., 2018), (Farzand et al., 2021) se define como una amenaza de la seguridad informática que tiene la especial peculiaridad de centrarse en una técnica basada en la observación sin consentimiento o también conocido como espionaje no autorizado de la información personal de la víctima donde principalmente se ve afectado la información referente a sus métodos de autenticación, de entre estos métodos de autenticación el que más se ve afectado es el método de autenticación basado en el conocimiento del usuario lo que es confirmado por (C. Wang et al., 2020) (Figura 1).

Figura 1

Representación de un ataque de Shoulder Surfing



Nota. La figura muestra la representación gráfica de como un atacante puede utilizar el ataque de Ingeniería Social “Shoulder Surfing” para robar información y este puede ser detectado por una video cámara integrada en el dispositivo.

Los atacantes o también denominados “Shoulder Surfers” conocidos así por la relación a este tipo de ataque de Ingeniería Social se han apoyado en las invenciones que se han implementado a lo largo de la Era Digital de tal manera que conforme al crecimiento exponencial de la cantidad de dispositivos móviles, servicios y aplicaciones web, los usuarios a lo largo del paso de los años han necesitado una forma para autenticarse, identificarse, diferenciarse y privar su información de los demás usuarios en el mundo, de modo que los usuarios accedan a todo tipo de cuentas de forma segura, cuentas que por ejemplo se las utiliza en el diario vivir; estas cuentas pueden ser bancarias, estudiantiles, laborales entre otros tipos de cuentas que son usadas para realizar movimientos monetarios, estudiar, trabajar, enviar correos, subir archivos, contenido multimedia, etc (Sun et al., 2018).

Los beneficios que obtienen los atacantes sobre las víctimas pueden ser monetarios hasta en algunos casos por motivos personales donde de alguna forma afectan a la víctima, según (Schneegass et al., 2022) los ataques de Shoulder Surfing (SSAs) ocurren con frecuencia en situaciones en las que la víctima y el atacante tienen una relación cercana e incluso se puede dejar abierta la posibilidad de darse en el campo laboral donde existe esta cercanía.

Escenarios de Shoulder Surfing

Al menos una vez se ha presentado la situación en la cual un amigo, compañero de trabajo, compañero de clase, personas en el transporte público, conocidos, o personas que comparten alguna actividad en donde el entorno y la cercana proximidad física interfieren han dado como resultado que se tenga que apagar la pantalla del dispositivo que se haya utilizado ese momento para evitar robo de información, debido a que una o varias personas han detectado que observan mensajes privados, notificaciones, redes sociales, etc., que según la información que visualicen y la cercanía relacional entre los actores del evento definen que tipo de mitigación el usuario propietario del dispositivo adopta para mitigar el Shoulder Surfing (Farzand et al., 2021).

Según (Brudy et al., 2014) menciona que a medida que las pantallas crecen en tamaño para las zonas semipúblicas y públicas se expone la información de sus actividades y datos sensibles de los usuarios a los transeúntes, lo cual representa un gran problema que puede ser solventada a través de un método de territorialidad y proxémica con el fin de regular comportamientos e interacciones sociales. Esto debido a que la gente por naturaleza tiene un sentido del espacio personal y la intimidad, de tal manera que es normal que las personas busquen proteger su propia información, sin embargo, esto no se da de la misma manera, ya que el factor personalidad y el contexto particular influyen en esto, esto se lo puede ver a través de la existencia de personas muy observadoras que utilizan las características espaciales para mantener controlado su espacio circundante e incluso en algunas ocasiones tienen la facilidad de modificar su entorno como se suele hacer en las oficinas personales al mover y posicionar cada uno de los muebles y dispositivos de oficina, sin embargo, esta ventaja se pierde en oficinas abiertas donde se tiene que cambiar posiciones de dispositivos de trabajo para tratar de alguna forma mitigar posibles ataques de Shoulder Surfing. La aplicación de estos conceptos ha llevado a que se tengan en cuenta dos puntos de vista que son la interacción

entre la conciencia además del protocolo social y la forma de posicionamiento del cuerpo de la persona para bloquear el campo de visión del atacante hacia el dispositivo, para entrar más en detalle el primero trata sobre el espacio personal o íntimo en la proxémica, esto resulta evidente que entre en juego debido a que es normal que el texto de un ordenador a la distancia sea imperceptible, de modo que el atacante se vea obligado a acercarse y entrar dentro del espacio personal de la víctima, el segundo punto de vista trata como el cuerpo puede resultar útil para ocultar el contenido a la vista de los atacantes, sin embargo, esto no es del todo posible en algunos casos debido al uso de varias pantallas de escritorio e inclusive esto empeora si estas son de un gran tamaño. A pesar de estos puntos de vista u formas de mitigación no son perfectas se puede rescatar la conciencia de los momentos de Shoulder Surfing y la protección de la información cuando se detecta el Shoulder Surfing.

Otros escenarios donde se suele dar este tipo de ataques son descritos en (Sikandar et al., 2019) donde no se describe al acto de Shoulder Surfing, sin embargo, sigue el mismo concepto en el cual los atacantes buscan sustraer información de credenciales de las víctimas mientras ocultan sus rostros con sombreros, gafas de sol, bufandas, máscaras, etc. de manera que no sean fácilmente reconocidos tanto por los transeúntes como por los dispositivos de video vigilancia que existen en los cajeros automáticos. Los atacantes también suelen instalar cámaras espía para captar los identificadores y contraseñas del usuario, otra forma también se da cuando los atacantes realizan actividades anómalas para crear alguna distracción en los cajeros automáticos con el fin utilizar dispositivos de skimming para la falsificación de tarjetas, y esto no es el único escenario donde los atacantes se valen de dispositivos u de algún objeto para obtener las credenciales de autenticación de las víctimas. En (Eiband et al., 2017) se menciona sobre algunos tales como las cámaras de vigilancia, reflejos por medio de la córnea, drones y cámaras de registros, aparte del convencional robo de contraseñas por observación directa.

Riesgos y consecuencias del Shoulder Surfing

Para (Eiband et al., 2017) en su investigación realizada con una muestra de 174 relatos de Shoulder Surfing se pudo determinar en dicho estudio que este ataque no suele tener consecuencias graves, sin embargo, evoca sentimientos negativos para ambas partes, adicionalmente se menciona que la información sujeta a Shoulder Surfing no fue restringida estrictamente a una autenticación, sino información visual privada general, principalmente se encontró que este tipo de ataques siguen un patrón determinado por la casualidad y la capacidad oportunista del atacante que en su mayoría fueron desconocidos mientras utilizaban un transporte público, durante los desplazamientos al trabajo, y las victimas hacían uso de su smartphone de tal forma que no tuvieron datos muestra de sucesos de carácter malintencionado con equipo técnico, a pesar de eso en dos de sus casos se encontró que los actores del suceso sintieron vergüenza, enfado o culpa y malestar, sin embargo, se detectó que solamente el 7 % de los incidentes fueron detectados por las víctimas las cuales trataron de utilizar diversas estrategias para apartar el dispositivo, proteger el contenido con el cuerpo o adaptar su forma de interactuar. A pesar de que existan varios estudios referentes al tema, en este documento se menciona que los riesgos reales del Shoulder Surfing aún no se han llegado a evaluar sistemáticamente y esto en parte debido a que el Shoulder Surfing no se limita únicamente a la observación de la introducción de contraseñas.

La investigación realizada por (Ponemon Institute, 2016) determinó que la viabilidad de los ataques de Shoulder Surfing en entornos de oficinas empresariales se incrementa en gran medida, de tal forma que se descubrió que gran parte de la información sensible de los usuarios se encontraba en dispositivos tales como ordenadores portátiles, tabletas y smartphones e inclusive en documentos o papeles que se dejan en escritorios, impresoras, mesas de conferencias entre otros lugares de la oficina o fuera de los sitios de reunión que resaltan a la vista de los Shoulder Surfers, además como datos importantes de esta

investigación que han realizado han encontrado que el 91% de los ataques tuvieron éxito debido a lo mencionado anteriormente, aunque el 12% de la información observada se basaba en credenciales de autenticación, otro dato importante que también señalan es que el 28 % de los ataques se realizaron en pantallas de ordenador desprotegidas.

En (Sun et al., 2018) se puede encontrar también que las personas pueden iniciar sesión en servicios web y aplicaciones en público para acceder a sus cuentas personales con sus teléfonos inteligentes, tabletas o dispositivos públicos, como los cajeros automáticos de los bancos, por lo que depende de las cuentas y al tipo de información a la que acceden, los atacantes de Shoulder Surfing pueden generar un alto o bajo impacto en la víctima, como por ejemplo el acceso a cuentas bancarias, cajeros automáticos en donde el riesgo incrementa por el hecho de que el atacante puede obtener la información bancaria y realizar transacciones u retiros donde la victima podría tener los ahorros de toda su vida, otro de los casos que puede suceder es cuando se utiliza servicios web o aplicaciones dentro de una organización en donde un compañero por hacer una broma o por deseos malignos puede afectar el perfil de usuario por medio del envío de correos a los superiores con amenazas, insultos, entre otras cosas más, con el propósito de afectar a esa persona hasta el punto de que sea expulsada de la organización y esto de cierta forma es corroborado por (Schneegass et al., 2022) y (Ponemon Institute, 2016) donde el primero se menciona que los ataques de Shoulder Surfing (SSAs) "ocurren con frecuencia en situaciones en las que la víctima y el atacante tiene una relación cercana" con esto en mente se puede dejar abierta la posibilidad de que también se produzca en el campo laboral y académico como se lo ha mencionado con anterioridad, mientras que en el segundo se aclara que si es posible que se dé un ataque de Shoulder Surfing en las empresas.

Técnicas y herramientas utilizadas en el Shoulder Surfing

Con respecto de las técnicas y herramientas utilizadas en el Shoulder Surfing en (Farzand et al., 2021), (Brudy et al., 2014), (Sikandar et al., 2019), (Sun et al., 2018) y (Behera et al., 2018) se han encontrado algunas técnicas y herramientas que son utilizadas en donde tanto la observación como el uso de un dispositivo entra en juego por lo cual los atacantes también han tratado de perfeccionar y evolucionar sus formas de hacer de las suyas de una forma más segura con el propósito de minimizar la exposición que puedan tener, en este caso la presencia del atacante para la toma de apuntes o la memorización de credenciales de autenticación de la víctima lo cual se lo realizaba en el escenario convencional donde el atacante sustrae información de la víctima a través de la observación, el escenario que se presenta se basa en la utilización de dispositivos de grabación con el fin de sustraer de una forma más segura toda información personal útil de la víctima por medio de la grabación en vídeo de las sesiones de autenticación con el fin de procesarlas offline para conocer la contraseñas.

Las dos técnicas donde se resume o categoriza es en (Behera et al., 2018) donde se categoriza a los ataques por “observación visual” o por “grabación”, sin embargo, no se diferencia tanto con respecto a la clasificación que se propone en (Sun et al., 2018), ya que en esta se determina tres tipos de categorías que son: i) Tipo I: Ojos desnudos, ii) Tipo II: El vídeo captura todo el proceso de autenticación una sola vez y iii) Tipo III: El vídeo captura todo el proceso de autenticación más de una vez, en donde la categoría de Tipo I hace referencia a lo mismo que se trata en la categorización de “observación visual” mientras que en Tipo II y III se trata lo mismo que la categorización por “grabación” por lo cual se define a todo acto de Shoulder Surfing del tipo observación en donde el atacante entra directamente a intervenir en la situación donde usa como herramienta la observación sea directa o indirecta a través de algún objeto como vidrios, espejos, entre otros que puedan facilitar el robo de la información, y en la

categoría de grabación a toda herramienta que pueda transmitir la información personal de la víctima hacia el atacante en esta categoría por lo general se puede encontrar a dispositivos tecnológicos implicados tales como cámaras espías, cámaras de videovigilancia, drones, smartphones, entre otros dispositivos que sean capaces de transmitir imagen (Figura 2).

Figura 2

Modalidades implementadas en el Shoulder Surfing

Nota. El gráfico presenta las dos modalidades o categorías en las cuales se puede presentar el Shoulder Surfing. En el lado izquierdo se presenta SSAs por grabación y por el derecho SSAs por observación visual.

Estación de trabajo, posicionamiento del usuario y relación con el Shoulder Surfing

La forma de posicionamiento en la silla, las manos sobre el teclado, el monitor y el entorno de trabajo puede tener implicaciones en la Ingeniería Social y el Shoulder Surfing lo cual es sustentado en parte por (Ponemon Institute, 2016) donde se menciona que se incrementa la viabilidad de ocurrencia de un ataque de Shoulder Surfing en entornos de oficina, sin embargo, el hecho de no adoptar unas posturas adecuadas, además de no proteger la pantalla de miradas indiscretas, podría facilitar la incidencia de este ataque, por lo que para evitar que aumente la probabilidad de esto es necesario que se mantenga una postura ergonómica durante el trabajo frente a una computadora mientras se use o no información personal. El estudio de (Emerson et al., 2021) menciona algunas recopilaciones de datos referentes a como llevar una postura ergonómica y como tener una estación de trabajo para

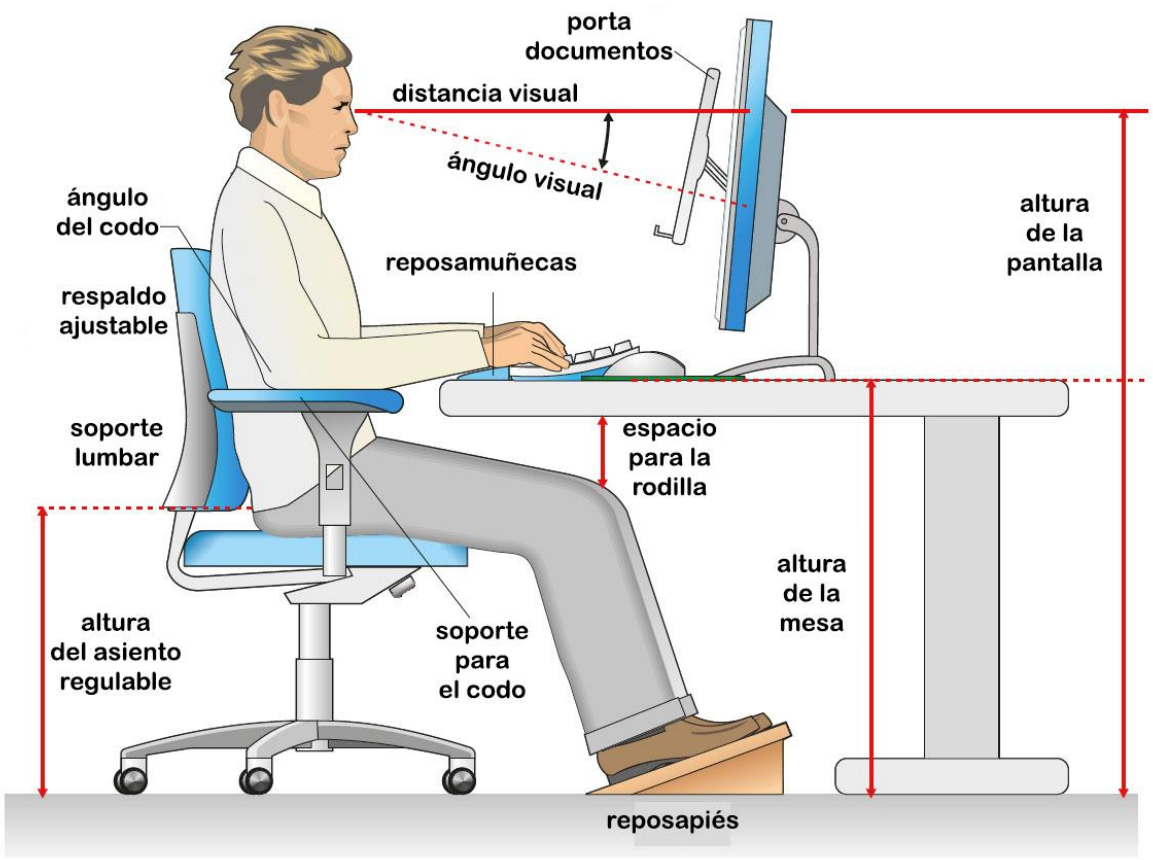
conseguirlo no obstante utilizar un estándar general para todos las personas con sus distintos tipos de cuerpo debido a la amplia variación de estatura, etnia, sexo, dimensiones antropométricas y sutiles diferencias entre los estándares internacionales, llegan a complicar las cosas por lo que se presenta de una forma general como llevar el puesto de trabajo, incluida la configuración, la postura general, los problemas de la silla, la posición y los tipos de teclado y ratón, las posiciones del monitor y el teléfono, el impacto de las gafas en la visión, la colocación de los documentos, los reposapiés y la iluminación, de tal forma que fortalece la comprensión de las interacciones entre los componentes y el trabajador.

Según (Emerson et al., 2021) el objetivo para conseguir una ergonomía en el posicionamiento de las personas en una silla debe lograr una postura neutra para el cuello, los hombros, los codos, las muñecas y la espalda, esta postura es conocida como postura sentada erguida, con los codos, las caderas, las rodillas y los tobillos colocados en ángulos de aproximadamente 90°, de tal manera que la parte superior de los brazos deba estar pegada al cuerpo, sin extenderse hacia delante ni hacia los lados, con los codos flexionados entre 70 y 90°, las muñecas neutras, la cabeza y el cuello erguidos con la mirada hacia delante. Ahora con respecto a la posición al teclado se menciona algunos descubrimientos en donde el uso de la posición de inclinación negativa de la bandeja para el teclado permite posturas más neutras para la muñeca y el codo, esta postura deberá seguir un ángulo para la muñeca > 30° para disminuir la presión del túnel carpiano si se teclea durante muchas horas, en otro estudio que se menciona en la misma investigación en cambio se recomendó que la extensión de la muñeca fuera < 15° para minimizar la tensión de la muñeca en donde a su vez de ser posible la persona pueda apoyarse con el uso de una bandeja de teclado, en caso de que el usuario utilice un reposapiés deberá hacerlo de una forma ergonómica en la cual la altura adecuada del reposapiés puede estimarse al simular la postura de cadera y la rodilla flexionadas 90°, por otro lado con respecto del monitor y de la distancia en la cual se debe observar la pantalla se señala

un rango de 63 a 85 cm, a continuación se coloca el tobillo como si estuviera en el suelo y se mide la distancia desde la superficie volar del pie hasta el suelo como se puede ver en la Figura 3. Además, en la misma investigación se señala que una estación de trabajo debe mantener adecuadas condiciones de luz y temperatura, esto también a su vez permite que el entorno de trabajo tenga las condiciones de luminosidad necesarias con el fin de que no afecte al reconocimiento facial y pueda crear falsos positivos.

Figura 3

Representación del posicionamiento ergonómico correcto para trabajar



Nota. El grafico muestra todos los accesorios, distancias y ángulos de una forma general, los cuales son necesarios para lograr una posición ergonómica correcta al trabajar frente a un ordenador. Tomado de Wikimedia Commons (MaxWellBorn, 2019).

La introducción de contraseñas a su vez como se ha tratado a lo largo de este trabajo resulta muy propensa a ser vulnerable ante ataques de Shoulder Surfing como lo asegura (Behera et al., 2018), donde adicionalmente en su investigación la cual buscó encontrar la robustez de una contraseña donde idearon un método para calcular la solidez de las contraseñas, en específico para contraseñas basadas en gestos o patrones, lo interesante fue que se realizaron grabaciones en vídeo de 840 sesiones de autenticación desde diferentes ángulos y posiciones con 20 participantes donde se encontró resultados los cuales evidencian que las contraseñas elegidas con cuidado dificultan un ataque de Surfing de tal modo que se obtiene una tasa media de verdaderos negativos del 94,4%. Lo interesante de esta investigación es que al realizar las grabaciones del ingreso de contraseñas indirectamente realizan un ataque de Shoulder Surfing a través del uso de la técnica de grabación en donde se utilizaron sensores y una video cámara junto con un trípode para alcanzar alturas de 5, 5.4, 5.8, y 6 pies de altura, se simulo la altura aproximada de un intruso y una altura de 5.5 pies para la víctima la cual se encuentra sentada frente al computador, todo esto dentro de un área rectangular de dimensiones 6 pies × 4 pies en donde se tomaron 7 posiciones enfrente con respecto a la pantalla del computador. Indirectamente no se toma importancia del posicionamiento que tiene el usuario de pruebas frente al computador cuando se realiza el experimento a pesar de ser un aspecto muy importante el cual como se ha mencionado por ejemplo en la sección de Riesgos y consecuencias del Shoulder Surfing donde se menciona que por naturaleza el usuario ante un evento de Shoulder Surfing busca proteger su información personal ocultándose con su cuerpo, esto lógicamente es posible realizar cuando se sitúa frente a un computador con el objetivo de reducir el ángulo de visión del atacante, esto se aplica cuando la víctima pueda percatarse del atacante, su ubicación y su ángulo de visión con respecto del dispositivo personal que observa, sin embargo, en el caso de que no se percate, entraría en juego el tema que se quiere tratar en esta sección lo que corresponde a las posiciones que una persona puede tener al utilizar un dispositivo.

Ataques relacionados con el Shoulder Surfing

A pesar de que los ataques de Shoulder Surfing sigan un definición fijada, existen veces en que se lo puede confundir con otro tipo de ataques, y en ocasiones existen conceptos y definiciones de otros tipos de ataques de Ingeniería Social que entran en juego, algunos de estos casos se los puede encontrar en (Sun et al., 2018) donde otros tipos de ataques se interceptan en un punto crítico el cual es representado por este problema con los ataques a los métodos de autenticación los cuales afectan a la seguridad informática y la privacidad que es otro punto que comparten estos ataques con el Shoulder Surfing. Los ataques que se presentan con cierta relación son ataques de mancha o también conocidos como “smudge attacks” (Sun et al., 2018) los cuales se basan en la extracción de información de los ingresos que se realizan en la pantalla táctil de un dispositivo por medio del uso del rastro de manchas dejado por huellas dactilares este ataque también se lo suele conocer como spoofing dactilar, el spoofing de reconocimiento facial (Cárabe & Cermeño, 2021) es un ataque el cual se basa en la utilización de máscaras 3D para suplantar la identidad de la víctima y obtener todo tipo de información que desee el atacante en todos los sitios que requieran un método de autenticación por reconocimiento facial, este tipo de ataque también tiene sus derivados los cuales se los habla en (Chan et al., 2018) donde se habla sobre el tipo de ataque por fotografía y por video donde básicamente son para el mismo método de autenticación que el anterior y se basan en poner una fotografía en un dispositivo lector de rostros para el reconocimiento facial o un video, el ataque del tipo morphing (Cárabe & Cermeño, 2021) el cual busca crear una imagen “morph” o transformada para alimentar el reconocimiento facial de dos se parezcan físicamente o tengan rasgos similares con el fin de que la persona A diferente de la persona B pueda hacerse pasar la verificación de reconocimiento facial por la persona de A hacia B, el skimming (Sikandar et al., 2019) es un tipo de ataque que busca clonar la información de las tarjetas, en algunas situaciones también se puede encontrar ataques híbridos los cuales utilizan dos o más tipos de ataques de Ingeniería Social para poder llegar a cumplir con su objetivo.

La detección facial y el reconocimiento facial

El reconocimiento facial para (Chan et al., 2018) se define como una de las características biométricas más comunes debido a que la información de la cara se puede extraer fácilmente sin la necesidad de ningún contacto físico, es también conocida por sus siglas al inglés FRT que quiere decir Face Recognition Technology o también en su forma reducida como FR Face Recognition. Para (Shao et al., 2021) el reconocimiento facial es una de las tecnologías más utilizadas en el campo del análisis y la identificación de rasgos fisiológicos humanos de entre los cuales es común ver aplicada esta tecnología en la verificación de la identidad de una persona ya que el rostro de una persona puede proporcionar información sobre su identidad o edad y sexo, sin embargo, el reconocimiento facial automático no se inició hasta la década de 1970 y su progreso se vio dificultado por las tecnologías en hardware de la época y no fue hasta la década de 1990 donde se desarrolló el hardware necesario para el correcto funcionamiento de esta tecnología. En esta misma década se vivió un proceso importante en el cual el reconocimiento facial que hace uso de imágenes como materia prima experimento cambios en diferentes fases de desarrollo conceptual que lo llevaron a dividirse en enfoques holísticos o basados en la apariencia que se centran en la región facial en general y métodos lineales o no lineales para mapear el rostro. En estos contexto el reconocimiento facial puede realizar tareas tales como conteo de personas para estimar el número de clientes que entran en una tienda y extraer algunas conclusiones estadísticas sobre sus características, por ejemplo, la edad o el sexo, la aplicación del método de autenticación facial que la mayoría de los teléfonos móviles del mercado llevan incorporada la cual permite desbloquearlos con una simple mirada al dispositivo (Cárabe & Cermeño, 2021). Algo que no debe de pasar por alto es que esta tecnología se la llegó a considerar tanto que ha llegado a utilizarse de manera plena por medio de su implementación en el desarrollo de aplicaciones relacionadas con la seguridad por sus beneficios (Shao et al., 2021).

El segundo término guarda cierta relación de por medio; sin embargo, no es lo mismo, ya que la detección facial va de la mano del reconocimiento facial, para ello se puede poner un ejemplo en el cual se requiere realizar un sistema para registro de entradas y salidas en una organización de forma automatizada, en este ejemplo se puede diseñar un sistema que funcione con detección facial para que pueda detectar los rostros de las personas, no obstante la pregunta y en si el objetivo a cumplir es generar registros de las personas en cuestión, cosa que solo al reconocer que una persona tiene una cara con ojos, nariz, boca, cejas y orejas no se puede dar, por esto se necesitaría una manera en la que la persona sea identificada y en este punto se puede ver la diferencia de estos conceptos ya que no necesariamente la detección facial tiene que cumplir con las condiciones de reconocimiento facial, ya que el uno en base a lo anteriormente visto detecta rostros, contornos y partes del mismo, tales como los ojos, nariz, boca, claro está que esto también dependerá del algoritmo con el cual haya sido construido, otro ejemplo claro del diario vivir para aquellas personas que les agrada la fotografía, es que en los dispositivos móviles inteligentes está integrado la detección facial al momento de fotografiar personas donde por lo general encierran con alguna figura la posición de los rostros de las personas, esto también se presenta en los dispositivos tales como los computadores. Según (Cárabe & Cermeño, 2021) encontró investigaciones donde se constata que cada vez más sitios web aplican políticas de "Conozca a su cliente" donde se compara una foto de identificación con una captura en tiempo real del solicitante que hace uso de este término de reconocimiento facial, otros estudios como (Chan et al., 2018) han encontrado que existen muchas aplicaciones de identificación personal las cuales ayudan al cumplimiento de la ley, la vigilancia, la seguridad de la información, la autenticación con tarjetas inteligentes y el entretenimiento, sin embargo, también se menciona que son vulnerable a ataques de suplantación de identidad.

Consideraciones que influyen negativamente al reconocimiento facial

A pesar de que en (Shao et al., 2021) se menciona sobre que el reconocimiento puede resistir cambios de luz, tono de piel, vello facial, peinado, gafas, expresiones y posturas, estos aspectos en conjunto con otros como en (Sikandar et al., 2019) que han sido mencionados con anterioridad donde existen atacantes que ocultan sus rostros con sombreros, gafas de sol, bufandas, máscaras, entre otros tipos de accesorios faciales que afectan a la precisión de la herramienta. En (Miraftabzadeh et al., 2018), (Hussain & Balushi, 2020) también se mencionan aspectos tales como el envejecimiento de los rasgos faciales, variaciones naturales del rostro, edad, escalas, poses, oclusiones, el cabello, el uso de accesorios, maquillajes e iluminaciones representan retos tanto para el reconocimiento facial como para la detección facial, estos retos representan un aumento a la probabilidad de fallo de la herramienta, sin contar claro está los ataques relacionados, en especial la suplantación de identidad que afecta tanto a esta temática como el Shoulder Surfing en sí.

¿Cómo intervienen las imágenes para la creación de videos y cuál es su representación en el computador?

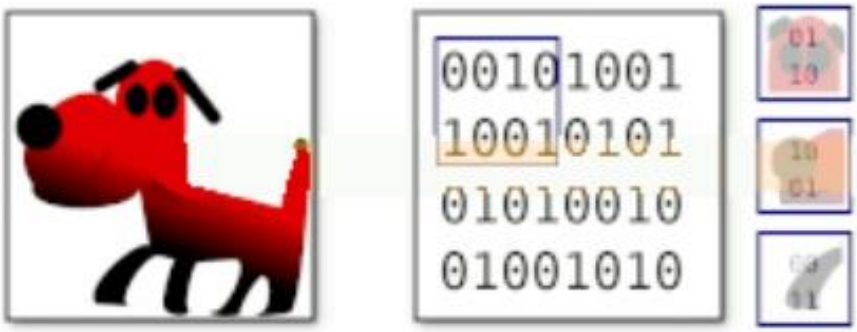
Para poder comprender cual es el significado o la importancia de las imágenes en un video primero se tiene que conocer el trasfondo y lo que implica una imagen, ya que un video no representa más que un conjunto de imágenes presentadas en el tiempo de forma sucesiva, la importancia que representa el conocer a profundidad lo que es una imagen, como está compuesta y cómo es que una computadora pueda llegar a entender que es una imagen se tiene que hablar de lo más básico, para lo cual (LeCun et al., 2015) habla sobre lo complicado que puede llegar a ser representar datos naturales en bruto a un lenguaje que pueda entender un computador, un claro ejemplo de esto puede ser el significado que tiene una palabra ya que por sí sola es un conjunto de caracteres o letras que no tienen significado y peor para un computador, no obstante gracias a los algoritmos de programación eso pudo llegarse a

entender, por ejemplo el código ASCII el cual es un código el cual se llegó a convertir en un sistema de codificación que asigna un valor numérico único a diferentes caracteres utilizados en la comunicación electrónica de tal manera que una computadora como originalmente se diseñó para entender números pueda llegar a entender y traducir, este caso es para que pueda una computadora entender que carácter queremos utilizar donde ingresamos un digito y retorna una letra, sin embargo, cómo puede llegar a entender una palabra un computador, para esto existen varios conceptos que se ven en la inteligencia artificial donde se tratan términos y algoritmos que se podrán conocer a profundidad durante el desarrollo del contenido, de momento se mencionaran de una forma más resumida que existen términos tales como embedding, distancias y algebra de vectores de palabras asociadas, donde el primero busca asignar un valor numérico a las pequeñas secciones de palabas o caracteres creadas, la distancia es la posición de cercanía que existe entre palabras en el espacio y finalmente la álgebra de vectores de palabras hace que se pueda encontrar palabras a partir de las operaciones matemáticas con otras, todo esto como una breve descripción a breves rasgos de todo este proceso que es conocido como procesamiento del lenguaje natural o NPL por sus siglas en inglés (Wolf et al., 2020). Con el fin de no extenderse más en este amplio contenido, se retoma el tema principal sobre que significa o que representa una imagen en el computador pues es algo similar en donde a través de procesos matemáticos se elabora una matriz de pixeles en la cual tendrá en cada una de sus posiciones valores numéricos entre 0 (negro) a 255 (blanco) que en pocas palabras representa la cantidad de luz que recibe el pixel de modo que permite al computador entender que significan los números como se puede ver en la Figura 4, sin embargo, ¿cuál es la diferencia que existe entre una imagen a color y una a escala de grises?, esto se puede solventar de una sencilla forma y es imaginarse un espacio tridimensional o dibujarlo, en el cual una simple matriz en dos dimensiones bidimensional están dadas por filas y columnas, ahora se debería sumar a esta una matriz al frente y otra detrás, de tal manera que se tendría una matriz para R (rojo), una matriz para G (verde) y una matriz para

B (azul) que son los colores que se necesitan para recrear una imagen a color y si a estos los juntas tendrías un vector de 3 elementos para cada celda o vista desde otra perspectiva se puede decir que son “tres matrices 2D que contienen las intensidades de los píxeles en los tres canales de color” como se menciona textualmente en (LeCun et al., 2015).

Figura 4

Procesamiento de imágenes; relación con pixeles y matrices



Nota. El grafico muestra como una imagen es entendida por un computador a través del uso de matrices de pixeles. Tomado de explainthatstuff (*How neural networks work - A simple introduction*, 2011).

Inteligencia Artificial

La inteligencia artificial como su nombre lo describe es una inteligencia que nace artificialmente, se define como la inteligencia que es dada o muestran las máquinas es por ello que se utiliza el término de artificial, en informática la inteligencia artificial también se le conoce como el estudio de los agentes inteligentes, de tal manera que sea llamado inteligencia artificial o agentes inteligentes son diseñados con el objetivo de cumplir alguna tarea de tal forma que se asemeje a las tareas que realizaría una persona, que por lo general lo programa de tal modo que busque maximizar la probabilidad de éxito en cumplir algún objetivo como los que se pueden nombrar a continuación: el razonamiento, el conocimiento, la planificación, el

aprendizaje, el procesamiento del lenguaje natural para la comunicación, la percepción y la capacidad de mover y manipular objetos, esto es posible gracias a que la inteligencia artificial puede llegar a imitar funciones cognitivas de los seres humanos tales como la función de aprender y solucionar problemas con algunos enfoques como estadísticos, la inteligencia computacional y la IA simbólica tradicional (Ongsulee, 2017).

Para (Janiesch et al., 2021) la inteligencia artificial es la capacidad de estos sistemas para resolver problemas avanzados, no obstante para poder lograr esto se basa en modelos analíticos que generan predicciones, reglas, propuestas, recomendaciones o resultados similares, con el fin de evitar posibles confusiones con los demás temas que se van a tratar. Es necesario que se aprenda bien las bases, definiciones y conceptos que abarcan todos los temas, para lo cual es necesario que se comprenda bien que es la IA para lo cual una idea general que resultará de mucha ayuda es tenerla presente como cualquier técnica que permita a los ordenadores imitar el comportamiento humano y reproducir o superar la toma de decisiones humana con el fin u objetivo de resolver tareas complejas de forma independiente o con una intervención humana mínima, no obstante la pregunta que salta a la vista de todos es ¿cómo el ser humano puede hacer aprender a una máquina cosas que ni él mismo entiende o en algunos casos también que es complicado explicar? y este es un paradigma que se enfrenta a varias limitaciones que se ven reflejadas en las limitaciones que existen todavía para que las inteligencias artificiales puedan realizar cualquier tipo de tarea compleja, esta idea o definición es clara y prácticamente igual a la que se manejó en el anterior párrafo, por lo que se espera que sea una ayuda para entender que es una inteligencia artificial y cuáles son sus limitaciones.

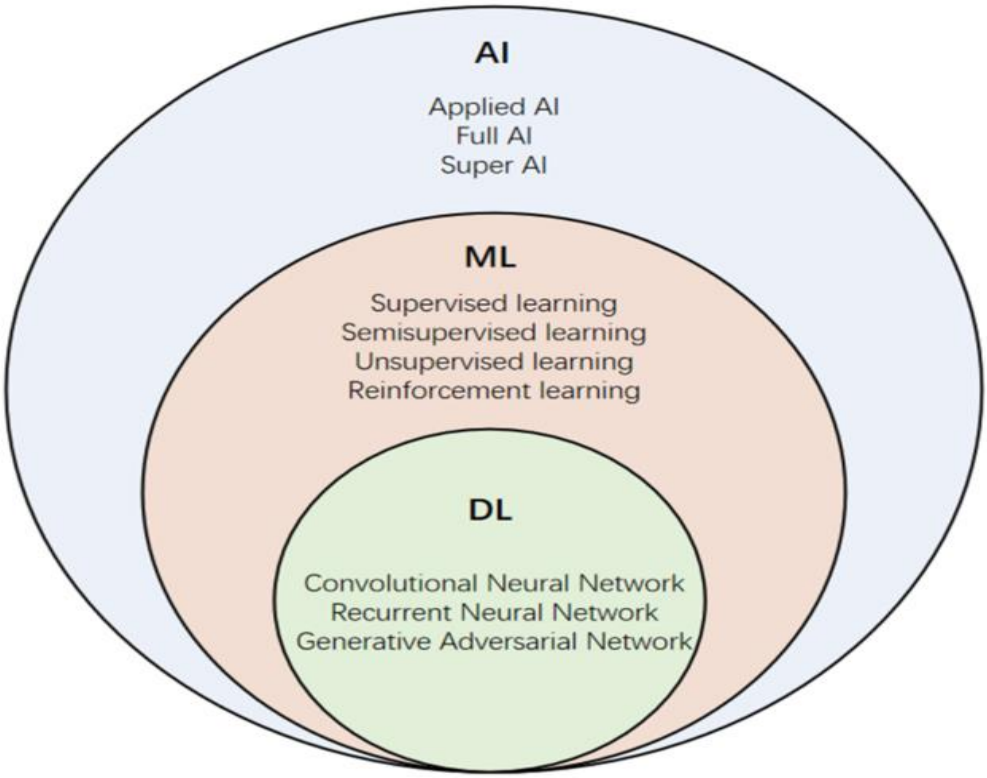
Machine Learning

Después de haber comprendido que representa la inteligencia artificial y cuál es su limitación se puede comprender lo que es el Machine Learning, ya que de una u otra forma esta

parte de la inteligencia artificial como se muestra en la Figura 5, es una parte de la inteligencia artificial que busca dar solución al limitante que se habló con anterioridad el cual es como poder hacer aprender a una máquina algo que resulta complicado de explicar para una persona la respuesta a esto es la experiencia y esto puede ser otorgada a las máquinas mediante algoritmos iterativos que aprende a partir de los datos de forma que la máquina realiza entrenamientos del problema y encuentra patrones complejos y percepciones que le ayudan a solventar una problemática compleja sin la necesidad de ser programados, esto ha permitido que la máquina pueda tomar decisiones fiables a partir del análisis de cantidades exhaustivas de datos y es por ello que se las ha llevado a realizar tareas complejas como análisis de la mejor oferta, detección de fraudes, para predecir los mercados de valores, para comprender las percepciones de los clientes, para analizar las necesidades de los clientes, para buscar productos, para el reconocimiento de voz, reconocimiento de imágenes y procesamiento del lenguaje natural (Janiesch et al., 2021). En (Ongsulee, 2017) se define que el Machine Learning es un subcampo de la informática y se basa principalmente en obtener como resultado un computador el cual haya aprendido alguna tarea sin la necesidad de haberle programado, esto es conseguido gracias al reconocimiento de patrones en muestras de datos en donde se aprende a partir de los mismos y se realiza predicciones sobre ello, para este estudio se menciona algunas aplicaciones en donde se puede evidenciar el uso del Machine Learning estas son el filtrado de correo electrónico, la detección de intrusos en la red, el reconocimiento óptico de caracteres (OCR), el aprendizaje de la clasificación y la visión por ordenador (CV).

Figura 5

El Deep Learning y su lugar dentro del universo de la IA



Nota. La figura plasma la posición en la cual se encuentra el Deep Learning dentro de las áreas de conocimientos de manera simple y resumida. Tomado de Wikimedia Commons (BrunelloN, 2021)

Métodos del Machine Learning

Debido a la gran diversidad de problemáticas que se quiere abarcar la implementación del Machine Learning y cada uno se resuelve de una distinta manera se han tratado de dividir en aprendizaje supervisado, aprendizaje no supervisado, aprendizaje semi supervisado, aprendizaje por reforzamiento, aprendizaje por transferencia y el aprendizaje profundo o también conocido como Deep Learning, según (Ongsulee, 2017) menciona que la frecuencia de uso de estos métodos va de la siguiente forma alrededor del 70% es supervisado, el aprendizaje no supervisado representa entre el 10 y el 20 por ciento mientras que el aprendizaje semi supervisado y el aprendizaje por refuerzo son otras dos tecnologías que se utilizan a veces.

Aprendizaje Supervisado

Para que se pueda utilizar este método de aprendizaje es necesario que se disponga de lo necesario, para lo cual en este caso se requiere datos tanto de entrada como de salida en donde estos datos de salida deberán tener etiquetas o valores objetivo de tal manera que cuando se quiera obtener un resultado de una entrada que no haya sido entrenada, se lo pueda hacer, esto resulta muy útil para resolver algunos problemas como la cantidad de me gusta en un video según el número de visitas en donde la variable y sería el resultado de me gustas del video mientras que x representa el número de visitas que ha tenido el video, supongamos que es un canal de videos musicales de una banda de música en particular y ha lanzado un nuevo hit, aunque resulte complejo determinar cuál es el nivel de agrado del nuevo disco por parte de la comunidad gracias a los datos obtenidos de anteriores discos se puede calibrar los paramentos abiertos de este modelo, para este y otros tipos de problemas que se puede presentar el Machine Learning como solución, ya que este tipo de aprendizaje funciona muy bien debido a que puede predecir con facilidad resultados de tal manera que a un valor de entrada le pertenezca un valor de salida esto visto en problemas de regresión también donde se predice valores numéricos y problemas de clasificación (Janiesch et al., 2021).

Aprendizaje No Supervisado

Este tipo de aprendizaje se da cuando el propio sistema de aprendizaje debe detectar patrones sin etiquetas ni especificaciones preexistentes de tal manera que los datos de entrenamiento solo constaran de variables x las cuales deberán descubrir cuál es la estructura de interés que se adapta al problema que se presenta, para tener más claro el campo de implementación de este aprendizaje se lo puede dar en un ejemplo en donde se agrupa datos similares en grupos o clústeres sin la necesidad de etiquetas o categorías previas donde se puede agrupar en función de la similitud de sus hábitos de gasto e ingresos donde se puede

aplicar un algoritmo popular llamado K-Means el cual es perfecto para entender este tipo de aprendizaje (Janiesch et al., 2021).

Aprendizaje Semisupervisado

Es un aprendizaje el cual sigue los mismos conceptos que el supervisado, excepto que este puede utilizar datos etiquetados como no etiquetados para su entrenamiento que por lo general son una gran cantidad de datos no etiquetados y una pequeña cantidad de datos etiquetados, la utilidad e importancia de este aprendizaje es porque los datos no etiquetados cuestan menos recursos el adquirirlos a diferencias de los datos etiquetados, al igual que el aprendizaje supervisado este aprendizaje sirve para solventar problemáticas relacionadas con la clasificación, regresión y la predicción (Ongsulee, 2017).

Aprendizaje por Reforzamiento

El aprendizaje por reforzamiento es un tipo de aprendizaje recurrente y en ocasiones muy utilizado por los seres humanos, en este se puede ver la implementación del principio de ensayo y error para encontrar resultados, para este aprendizaje se debe de especificar el objetivo junto con una lista de acciones y restricciones para cumplir con el objetivo por ejemplo se busca crear un ratón robot y se lo introduce en un laberinto con el objetivo de salir de dicho laberinto, el ratón tendrá la restricción de no saltarse las paredes del laberinto y como acciones a poder realizar es moverse en la dirección al frente, izquierda, derecha y atrás, de tal manera que el ratón aprenderá la ruta de salida por prueba y error a medida que logre salir del laberinto una y otra vez hasta encontrar la forma óptima de salir del mismo (Janiesch et al., 2021).

Aprendizaje por Transferencia

Por lo general el entrenar modelos completos de Machine Learning o Deep Learning son costosos debido a que requiere una abundante cantidad de datos, no obstante es allí donde entra en juego este tipo de aprendizaje, ya que este aprendizaje no requiere entrenar modelos desde cero y a su vez permite utilizar modelos que hayan sido entrenados por

conjunto de datos generales, un claro ejemplo puede ser que a un modelo se le entre con un conjunto de imágenes de animales, entre los cuales se encuentra un subconjunto de animales domésticos, en este punto lo más general puede transferir el conocimiento a otro modelo que busque animales domésticos de crianza, a través de la herencia de las características aprendidas con anterioridad es posible transferir dicho conocimiento, sin embargo, no es del todo recomendable ya que si se realiza esto existe una caja negra la cual se desconoce completamente su funcionamiento esto implica sesgos, ataques adversarios lo que abre puertas a limitaciones del entorno o que contengan puertas traseras que produzcan errores en la clasificación de imágenes por ejemplo, otro factor que puede marcarse en este proceso son las intervenciones gubernamentales para redirigir o suprimir predicciones, por lo que este tipo de aprendizaje se puede utilizar, sin embargo, para ello se necesitará correr posibles riesgos o partir de modelos de completa confianza e implementar nuevos modelos a partir de ellos en sistemas inteligentes para las organizaciones en aplicaciones específicas (Janiesch et al., 2021).

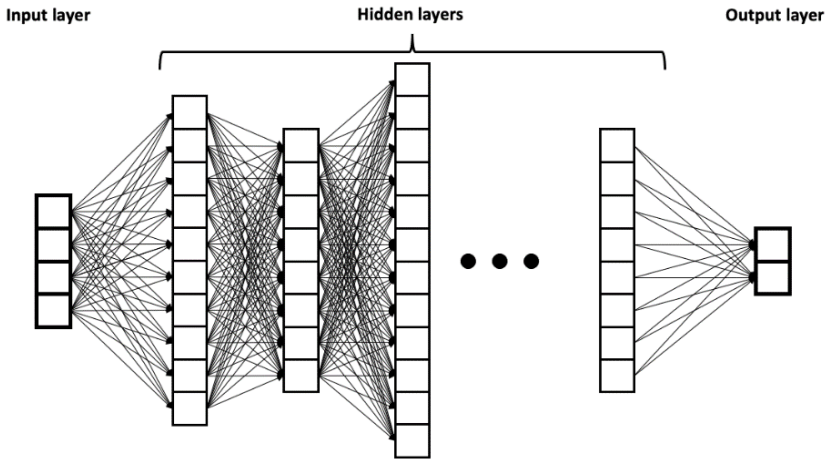
Aprendizaje Automático Profundo (Deep Learning)

Aprendizaje profundo, aprendizaje estructurado profundo, aprendizaje jerárquico, aprendizaje automático profundo, son algunos nombres que se les ha dado a este tipo de aprendizaje el cual se basa en el estudio de redes neuronales artificiales, para tener una idea clara y que se pueda comprender este tema, una red neuronal es un modelo matemático y a la vez computacional el cual ha sido diseñado o inspirado en la estructura y funcionamiento del cerebro humano enfocándose en cómo el ser humano puede generar conocimiento a través del procesamiento e intercambio de información entre neuronas también conocido como sinapsis. Una vez entendido esto, se retoma el tema principal donde se deberá señalar que este proceso anteriormente explicado guarda esa misma similitud para el intercambio de información entre cascadas de capas sucesivas las cuales utilizan la salida de la capa anterior como entrada con

el fin de transformar y extraer características. El hecho de que también se le llame a este aprendizaje como jerárquico es debido a que las características de nivel superior se derivan de las de nivel inferior para formar una representación jerárquica como se puede ver en la Figura 6, cabe recalcar que el Deep Learning al ser un aprendizaje que trata con redes neuronales, abre el paso a varias arquitecturas las cuales algunas son redes neuronales profundas, redes neuronales profundas convolucionales, las redes neuronales profundas de creencia y las redes neuronales recurrentes las cuales se han diseñado explícitamente para resolver algunas problemáticas de tal manera que se ha llegado a aplicar en campos como la visión por ordenador, el reconocimiento automático del habla, el procesamiento del lenguaje natural, el reconocimiento de audio y la bioinformática que llega a demostrar resultados satisfactorios (Ongsulee, 2017). Como se comentó con anterioridad que el uso de las redes neuronales ha permitido que se utilice en aplicaciones tales como la visión por ordenador a continuación se presenta un ejemplo claro de esto el cual sería al entrenar una red neuronal para distinguir la anatomía del rostro humano, en donde las capas aprenderán desde los más básico como la distinción de líneas y contornos con el fin de determinar figuras de las que está compuesto el rostro humano hasta determinar figuras compuestas, entre otras características tales como si son estáticas, o tienen movimiento y demás características que podrían presentarse, para que se tenga más claro esto se puede tomar el ejemplo del iris en el ojo humano. En fin una de las cualidades más interesantes que se puede resaltar de las redes neuronales son excelentes para “procesar datos que vienen en forma de matrices múltiples”, debido a que son el aporte perfecto para el tratamiento de imágenes lo cual llega a ser un tipo de aprendizaje perfecto para aplicar en tecnologías de reconocimiento facial (LeCun et al., 2015).

Figura 6

Estructura general de las redes neuronales



Nota. La imagen muestra la estructura básica de una red neuronal que a su vez representa un modelo computacional para el trabajo del Deep Learning. Tomado de Wikimedia Commons (BrunelloN, 2021).

Redes Neuronales

Las redes neuronales son un modelo que simula las redes neuronales que tenemos en nuestro cerebro y que ayudan a poder tomar decisiones y tener la capacidad de poder aprender en base a estudios y experiencias. Lo que se busca con estas redes artificiales es que tengan la capacidad de aprender en base a una información principal (de entrada) y que sirve para poder producir un resultado que será utilizado como una próxima información de entrada. La neurona es el elemento fundamental de procesamiento de nuestro sistema nervioso y que interconectadas entre sí se vuelven un sistema muy potente, la inteligencia que mueve al ser humano, literalmente.

La neurona humana actúa mediante estímulos que son producidas por entradas, estas son obtenidas a través de los sentidos, cuando alcanza un estímulo alto esta se activa, y proporciona una señal que será trasladada a otra neurona, lo que permite producir una sucesión de activaciones que desembocan en una decisión u acción que realiza el ser humano.

Las redes neuronales artificiales o por sus siglas en inglés ANN (Artificial Neural Network) están formadas por neuronas entrelazadas y que constituyen capas, estas capas se pueden diferenciar en capa de entrada que será la parte en donde se ingresa toda la información. Luego siguen capas ocultas que serán las encargadas de utilizar las respuestas de las capas principales para hacer más procedimientos de toma de decisiones y según su complejidad tendrá distintos diseños o topologías. Por último, la capa de salida que presenta los resultados que se obtuvieron en las capas anteriores (Ruiz et al., s. f.).

Las ANN pueden clasificarse por su topología en la que se tiene la Red Feedforward y Backforward, estas son redes que van hacia adelante hasta tener una salida, es decir que son acíclicas, no retornan y no tienen ciclos dentro de la misma capa, por otra parte, existen las monocapa, multicapa y red recurrente, estas identificadas por su estructura. La segunda clasificación es por su algoritmo de aprendizaje, subdivide en 4, el primero es el aprendizaje supervisado el cual debe tener una datos preclasificados o que su salida ya sea conocida, el segundo es el aprendizaje auto organizado, es decir que los datos no necesitan estar ordenados, el tercero es las redes híbridas que es la combinación del supervisado y auto organizado, puede tener datos de los 2 tipos, y por último tenemos el aprendizaje reforzado que se encarga de potenciar los dos primeros modelos (*267928931.pdf*, s. f.)

Cada que avanza la tecnología surgen más necesidades y requerimientos, las ANN han sido aplicadas a varias problemáticas para ayudar a resolver estos problemas y darles distintas aplicaciones. Algunas de ellas son la conversión de texto a voz, el procesado natural del lenguaje, reconocimiento de caracteres, compresión de imágenes, filtro de ruido, modelado de sistemas y reconocimiento de patrones de imágenes. Se hace hincapié en esta última aplicación, la detección de imágenes ha sido utilizado para potenciar la visión por computadora para enseñar al equipo a entender lo que se le muestra, en la parte de la medicina ayuda a dar un diagnóstico por el análisis de las imágenes del paciente y de los exámenes que se realizan,

en la seguridad física ayuda a interpretar las actividades y comportamientos sospechosos, en el entretenimiento se ha aplicado en la detección de imágenes para poder aplicar realidad aumentada, realidad virtual, filtros de imágenes y por último el reconocimiento facial que ayuda a potenciar la seguridad y garantizar la autenticidad del propietario de dispositivos o programas (*libro-del-curso.pdf*, s. f.).

Las Redes Neuronales Artificiales progresan día a día para poder obtener resultados satisfactorios y que beneficien al ser humano, cada día su uso toma un papel protagónico en la creación y diseño de nuevas tecnologías. Por lo que tener un conocimiento básico de que son y cómo funcionan es esencial para estar en la vanguardia de la ciencia informática.

Estado del Arte

Factibilidad e impacto de la implementación del reconocimiento facial en la detección y mitigación de ataques de Shoulder Surfing

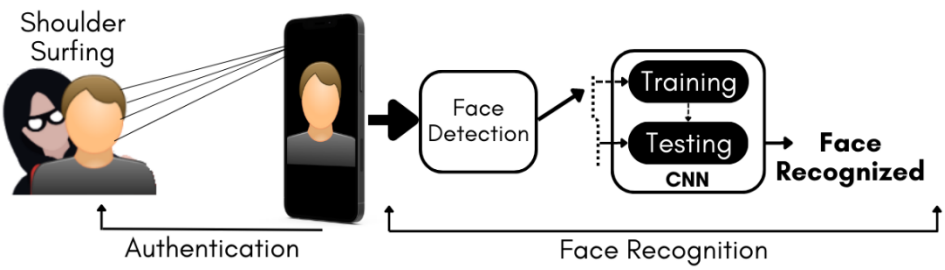
El presente tema busca dejar plasmada la importancia y la base de conocimiento que permite amparar la realización de este trabajo, para lo cual es necesario que exista unas bases sólidas en las que se pueda sentar todo el proceso que tiene que ver con la detección y mitigación de ataques de Shoulder Surfing, es por ello que a continuación gracias al correspondiente estudio del arte es posible tener unas bases sólidas las cuales ayudarán a determinar el futuro que le depara al desarrollo de la aplicación que se propone con el objetivo de señalar cuán factible puede ser el aplicar las tecnologías que se planea utilizar para solventar posibles problemas de la seguridad de la información.

Según (Chan et al., 2018) se puede ver a breves rasgos como la implementación del reconocimiento facial es aplicado para solventar problemas referentes a la identificación personal con el fin de dar cumplimiento a la ley, la aplicación en la seguridad en vigilancia y la implementación en general de esta tecnología para ayudar a fortificar la seguridad de la

información, esto también guarda la misma relación con lo que se menciona en (Hussain & Balushi, 2020) donde señala que esta tecnología se ha aplicado en la seguridad, vigilancia con cámaras, verificación de identidad en dispositivos electrónicos modernos, investigaciones criminales, sistemas de gestión de bases de datos y aplicaciones de tarjetas inteligentes, etc. Para (Shao et al., 2021) se pudo encontrar que el reconocimiento facial tras el progreso que tuvo esta tecnología en los años posteriores a la década de 1990 se tomó esta tecnología para implementarla en aplicaciones relacionadas a la seguridad. Como se pudo constatar en estos estudios se habla sobre esta tecnología asociada con implementaciones en seguridad donde a su vez guarda esta relación con los métodos de autenticación como por ejemplo en (Cárabe & Cermeño, 2021) donde hace referencia a los dispositivos móviles que forman parte del diario vivir de las personas en los cuales hace tan solo simple una mirada del propietario para poder desbloquearlos, esto también es dado gracias al reconocimiento facial a través de la identificación personal, en este punto se puede observar cómo interactúa el reconocimiento facial sobre los métodos de autenticación y esto a su vez se enlaza con los conceptos anteriormente tratados del Shoulder Surfing, sin embargo, cómo es posible que se pueda detectar que una persona es o no un atacante, esto resulta algo sencillo en realidad de tal forma que tan solo la implementación de un identificador al dispositivo y este enlazarlo al propietario, en concreto al rostro e identificación personal del mismo permitiría un futuro exitoso para el desarrollo de la aplicación (Figura 7).

Figura 7

Detección y reconocimiento facial su proceso e importancia



Nota. El gráfico muestra el proceso de la detección y reconocimiento facial para encontrar atacantes en dispositivos personales a través de la video cámara integrada en el mismo.

Con el fin de retomar conocimientos necesarios para continuar esta sección se hará un breve recordatorio sobre las técnicas utilizadas para efectuar un ataque de Shoulder Surfing se puede encontrar dos escenarios en los cuales, el primero se presenta cuando el atacante utiliza la observación para el robo de información, en este caso es posible la identificación a través del reconocimiento facial del posible atacante o el usuario ajeno a ser el propietario del dispositivo personal, no obstante para el segundo escenario se presenta una interesante pregunta la cual es ¿qué sucede cuando entra en juego el dispositivo electrónico que graba el proceso de introducción de la contraseña? Para este caso el objeto a ser reconocido no es una persona como tal a modo de que el convencional modelo de reconocimiento facial aplicado a la identificación de personas no pueda encontrar atacantes. A pesar que resulte complejo la ejecución del segundo escenario debido a la complejidad y aun existente riesgo del atacante a ser rastreado, se requiere más estudio de la víctima, planeación, esfuerzo, y recursos de todo tipo, incluido los recursos necesarios para la adquisición de dispositivos lo suficientemente potentes para que se pueda realizar esta técnica de Shoulder Surfing a comparación de la tradicional que resulta más sencilla y con menos costo en recursos, sin embargo, tampoco se puede dejar de lado la posibilidad de que se pueda presentar de esta forma el ataque, por lo cual es necesario que se analice detalladamente como el reconocimiento facial o el algoritmo y su composición pueden llegar a identificar rostros, a fin de determinar la posibilidad de que exista un reconocimiento capaz de identificar y cubrir ambos casos. El reconocimiento facial parte de las redes neuronales en especial y con resultados satisfactorios en las convolucionales CNN por medio del aprendizaje profundo a través de un modelo basado en algoritmos que llegan a obtener una representación compacta y separada del ruido que pudiese presentarse en la comparación de imágenes de rostros (Shao et al., 2021), este concepto

permite asegurar parcialmente que existe la posibilidad de entrenar redes neuronales para detectar ambas cosas tanto rostros como dispositivos electrónicos de grabación y esto se lo puede confirmar en su totalidad gracias al estudio de (Toor et al., 2018) en donde se habla sobre una investigación en el cual las CNN a través de la visión computarizada CV permiten la detección/reconocimiento de objetos genéricos y atributos biométricos. Gracias a esto y las distintas características de los dispositivos electrónicos es posible que se pueda generar procesos de mitigación de SSAs como por ejemplo atenuación del brillo del dispositivo, bloqueo automático del dispositivo, despliegue de alertas a la víctima para que tome medidas de precaución, etc.

Artefactos y herramientas que han sido utilizados para detectar y mitigar los ataques de Shoulder Surfing

En el proceso de detección y mitigación hace parte fundamental la lógica del artefacto de software, en donde se encuentra el diseño de la interfaz de usuario, el código y algoritmos de procesamiento de aprendizaje automático como las redes neuronales convolucionales, especialmente cuando se tiene de por medio el proceso de reconocimiento de imágenes (Tu et al., 2019). Durante la evolución de los procesos lógicos, el código también sigue retroalimentándose, como se explica en el estudio de (Z. Wang et al., 2023)**,** que se utiliza una red convolucional de diferencia central (CDCN), que además de cumplir con las funciones de una red convolucional normal, puede ahora recolectar la textura y predecir la profundidad de la cara.

En la parte física se han utilizados muchas herramientas como se menciona en el estudio de (Schneegass et al., 2022), en donde se diseñó un estuche para smartphones que incluye un lente de ojo de pez en la cámara frontal que amplía el campo de visión de la cámara de 60 a 180 grados, lo que permite obtener una imagen clara del atacante. En el estudio de (Brudy et al., 2014) hace énfasis en el movimiento, por lo cual se usa una gran pantalla y un

sistema de seguimiento llamada Vicon, para poder identificar la posición de cada persona, específicamente de su torso y su rostro, esto con ayuda de sensores. En la Figura 8 se pueden observar los mecanismos físicos que se explicaron brevemente. Sin embargo, por su alto costo otros estudios han optado por tener herramientas más accesibles económicamente, como Kinect o cámaras de computadores de escritorios o las que ya vienen adheridas en el caso de laptops, como se evidencio en el estudio de (Qin et al., 2021).

Figura 8

Herramientas utilizadas para la detección del Shoulder Surfing

Nota. La figura muestra en la parte de la izquierda la pantalla y el sistema de movimiento, mientras que en la derecha el smartphone con el ojo de pez.

Impacto del Shoulder Surfing en los métodos de autenticación

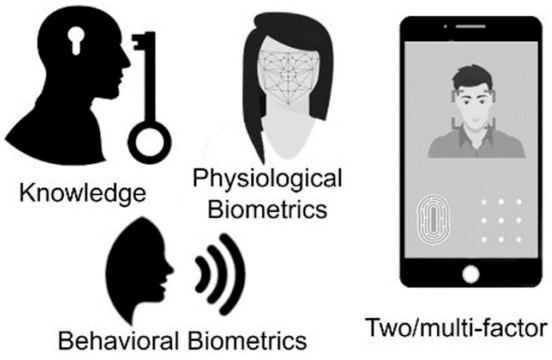
Pese a que un ataque de Shoulder Surfing afecte solamente a métodos de autenticación sino a información visual privada que también es mencionado en (Eiband et al., 2017), por lo general la información privada con gran importancia y afectación es la referente a la autenticación, por lo que el avance de los métodos de robo de información de credenciales ha obligado a las tecnologías de la información de cierta forma a fortalecer las brechas de seguridad de sus métodos de autenticación, con lo cual se han desarrollado una gran cantidad de formas de autenticación como por ejemplo las que se mencionan en (Ibrahim et al., 2019) donde los autores categorizaron 56 formas de autenticación en las siguientes categorías: en

métodos de autenticación basados en PIN, método de autenticación basado en la biometría, autenticación basada en patrones/gesto, autenticación de contraseña gráfica y otros métodos de autenticación, además los autores analizaron estas formas de autenticación según la existencia o no de afectación de los siguientes ataques: ataques de adivinación, ataque de diccionario, ataque Easily Stolen, suplantación de identidad, ataque de keylogger, ataque de mancha, ataques de fuerza bruta y la más importante para este trabajo la cual es Shoulder Surfing en donde se puede señalar que 20 de las 56 formas de autenticación si se muestran afectadas por SSAs. En otro estudio (C. Wang et al., 2020) se encontró una categorización de formas de autenticación de una forma más clara y precisa como se puede ver en la Figura 9, de tal manera que según dicha investigación se terminó por utilizar las cuatro categorías que se mencionan a continuación:

- Basada en conocimiento: Requiere que el usuario proporcione información que solo él debería conocer por ejemplo contraseñas, PIN de dígitos, alfanuméricos, patrón de bloqueo de contraseña, puntos de clic secretos en imágenes, respuestas a preguntas de seguridad o frases secretas.
- Basada en biometría fisiológica: Utiliza características físicas o comportamentales únicas del usuario por ejemplo Reconocimiento facial, Huella dactilar, voz, patrones de escritura e iris.
- Basada en biometría del comportamiento: Analiza datos relacionados con la forma en que una persona interactúa con un sistema o dispositivo por ejemplo Movimientos corporales, marcha y gesto de la mano
- Basada en dos/múltiples factores: Combina dos o más métodos de autenticación de diferentes categorías para proporcionar una óptima seguridad por ejemplo autenticación de dos factores (2FA) o la autenticación de múltiples factores (MFA)

Figura 9

Categorización de las formas de autentificación



Nota. La figura muestra una representación de las cuatro categorías de formas en las que se puede clasificar los tipos de autenticación de usuario UA.

De acuerdo con (C. Wang et al., 2020) y (Ali et al., 2021), la autenticación basada en conocimiento es ampliamente reconocida como el enfoque más popular y ampliamente utilizado a lo largo de la historia. Ha ganado una gran cantidad de seguidores y se considera el más efectivo en términos de obtener resultados positivos mediante “Ataques de Shoulder Surfing" (SSAs). Este hecho se ha respaldado en estudios anteriores como (C. Wang et al., 2020), (Toor et al., 2018), (Zhou et al., 2023) y (Ibrahim et al., 2019), donde se ha demostrado que es posible llevar a cabo un robo directo de credenciales a través de SSAs. En (Toor et al., 2018) se enfatiza que las contraseñas memorables, especialmente aquellas que involucran la selección de imágenes, son particularmente vulnerables a los SSAs, lo que facilita su recuerdo por parte de los atacantes.

Asimismo, (Zhou et al., 2023) destaca la creciente preocupación por el riesgo de los SSAs en dispositivos móviles, donde se considera la interacción del usuario, el ángulo de observación, el error de entrada y el esfuerzo de observación. En (Ibrahim et al., 2019) se concluye que ciertas formas de autenticación basadas en conocimiento son especialmente

susceptibles a los ataques de SSAs, lo que representa un desafío importante en términos de seguridad.

Reglas y parámetros para desarrollar artefactos de software centrados en la mitigación de ataques de Shoulder Surfing

Para poder desarrollar una herramienta de software que les haga frente a los ataques de Shoulder Surfing y pueda mitigarlos, hay que tener claro que se necesita proteger y no convertirse en un atacante. Al utilizar una tecnología de detección facial, se puede quebrantar la privacidad del atacante, al predominar su posición de ser humano y no de delincuente, aunque no tenga un sentido lógico (Schneegass et al., 2022). Capturar fotos de manera automática es una técnica inapropiada, por lo que una práctica coherente en este contexto es almacenar imágenes para la detección, sin embargo, no para la divulgación. Esto se puede ver por ejemplo cuando una persona sin tener malas intenciones puede ser captada por la cámara y así caer en posición de atacantes sin ser esto cierto.

Cada país tiene ciertas normas y leyes que se deben cumplir con respecto a la información de sus ciudadanos. En Ecuador la ley orgánica de gestión de identidad y datos civiles hace hincapié en que el Estado garantizará a todas las personas ecuatorianas y extranjeras, independientemente de su condición migratoria, el derecho a la identidad y protección de datos del personal de información (*LEY_ORGÁNICA_DE_GESTIÓN_DE_LA_IDENTIDAD_Y_DATOS_CI_751*, s. f.) **.** Dado esta normativa, la regla básica, fundamental e irrompible que se debe considerar durante el desarrollo del aplicativo es la privacidad del ser humano. Esto no limita que se pueda hacer la referencia al atacante, ya que exponer el acto sin exponer la identidad es algo que se puede realizar para alertar a la víctima.

Dada la premisa anterior, un estudio planteo una idea que consiste en degradar la calidad de las fotografías que se capturen en primera persona, esto con la ayuda de filtros de

desenfoque, para que las personas no se encuentren aludidas y observadas, para que puedan motivarse a participar de la captura de su rostro. Los resultados obtenidos reflejaron que el 17.5% de las personas incrementaron el deseo de ser captados por una cámara solo si se aplica el desenfoque de las imágenes capturadas (Dimiccoli et al., 2018).

En conclusión, se debe tener cuidado con vulnerar la integridad del ser humano, de manera especial porque no se conoce si esta persona puede ser peligrosa o no, ya que si no se considera los derechos de la humanidad se produce un cambio personal al estado de atacante que vulnera la información sin consentimiento.

Capítulo III: Diseño y Desarrollo

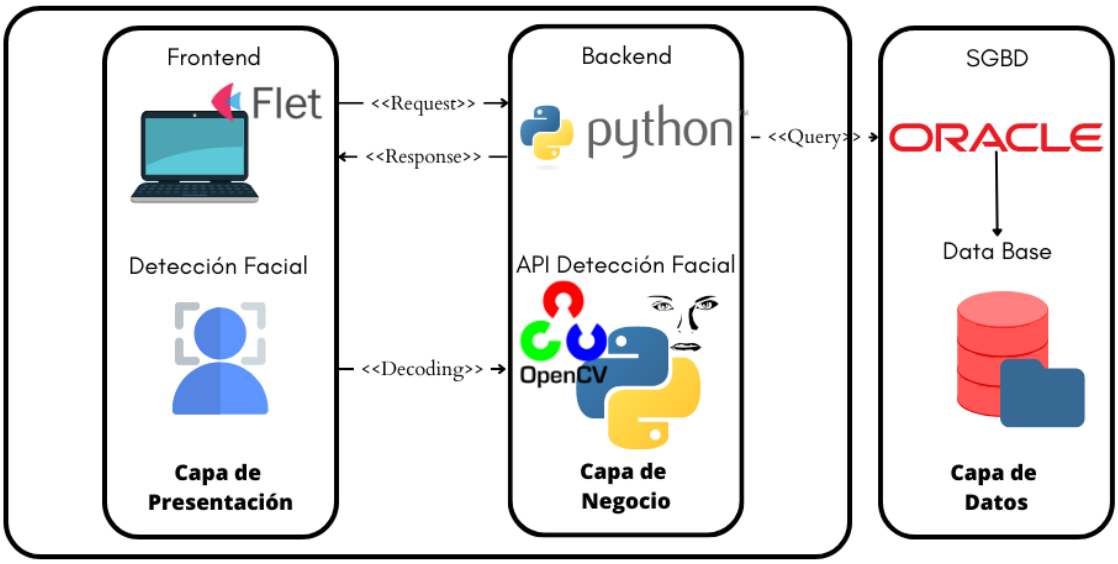
Fase de Diseño

En este apartado se detalla toda la construcción y diseño de la aplicación, desde su arquitectura de software, diagrama de procesos, diagramas de casos de usos, diagramas de clases, modelo lógico y físico de la base de datos, y diagramas de secuencia. Esta serie de gráficos reflejan el funcionamiento general y la estructura del artefacto de software.

Arquitectura de Software

Figura 10

Arquitectura de la Aplicación



Nota. El grafico muestra la arquitectura de software mediante una división de capas.

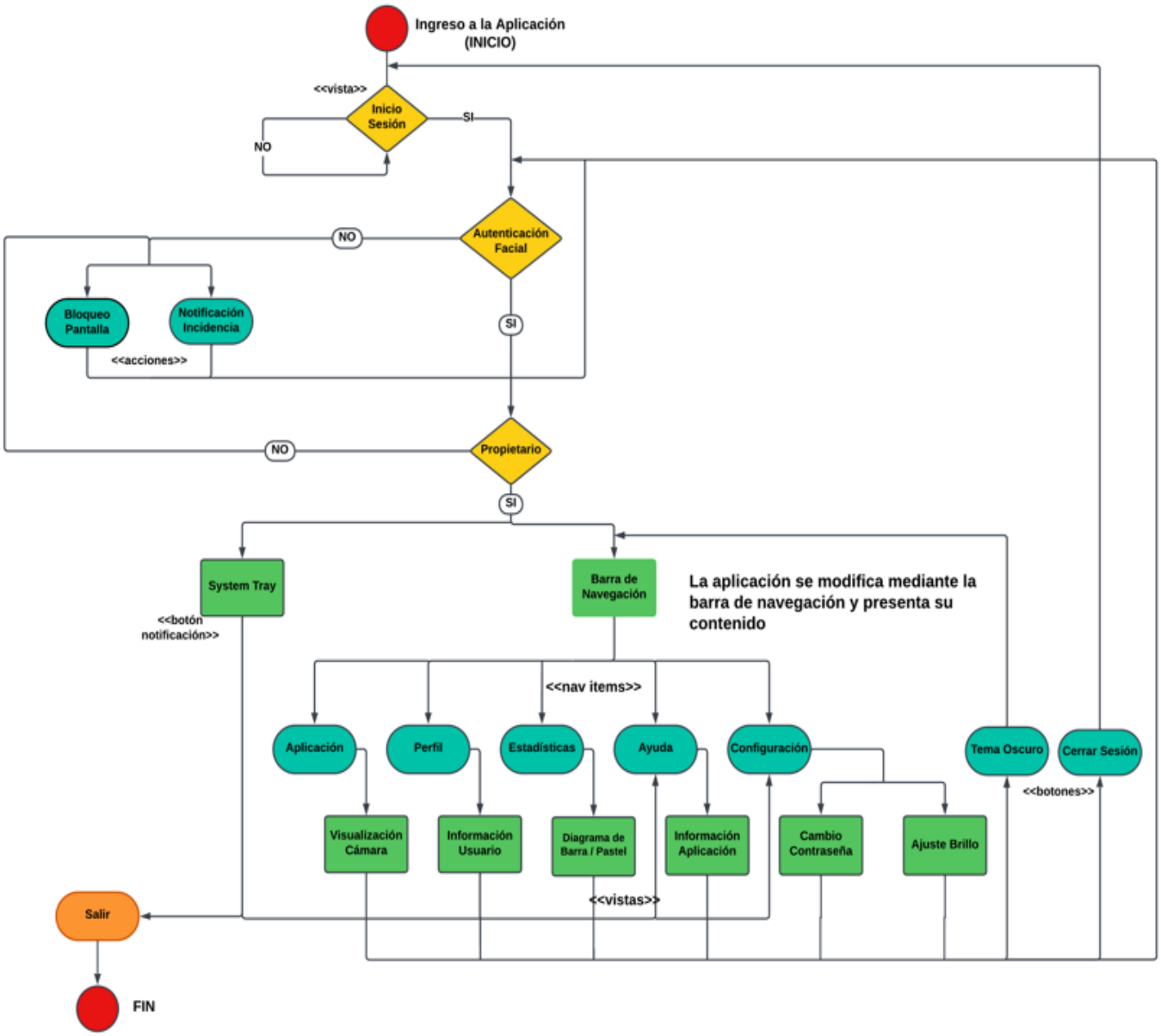
El modelo de arquitectura de software se divide en 3 capas; en el inicio se puede encontrar la capa de presentación, en donde se realiza la interacción con el usuario, mediante la creación de la interfaz con Flet y la detección facial mediante la cámara del dispositivo. La capa de negocio contempla la codificación que interactúa con la base de datos y con la interfaz, acompañada de la API de detección facial que tiene su cimiento en OpenCV. Por último, la

capa de Datos en donde se encuentra el sistema gestor de base de datos, en este caso Oracle y la base de datos que maneja.

Diagrama de Procesos

Figura 11

Diagrama de Procesos de la Aplicación

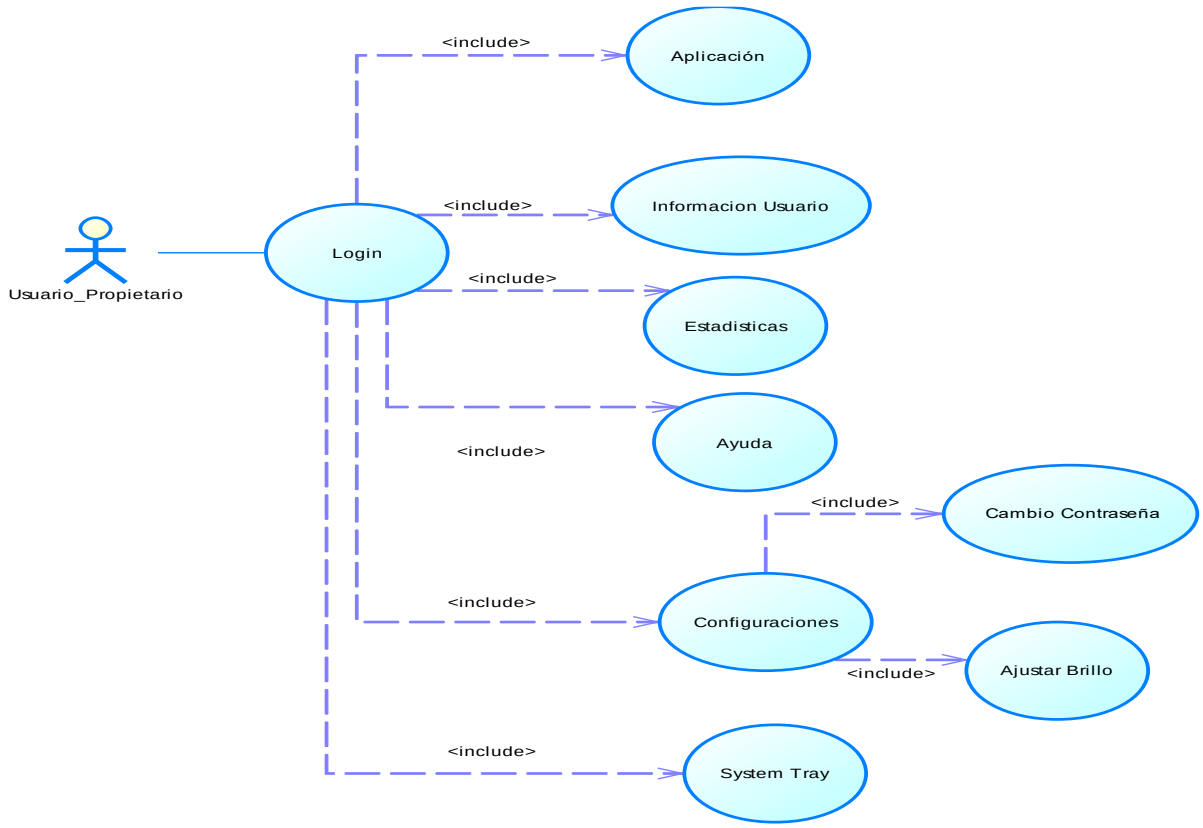


Nota. El grafico muestra el diagrama de procesos, con sus respectivos subprocesos y condiciones.

Diagramas de Casos de Uso

Figura 12

Diagrama de caso de uso de ingreso a la aplicación



Nota. El grafico muestra el caso de uso del ingreso de la aplicación

Tabla 1

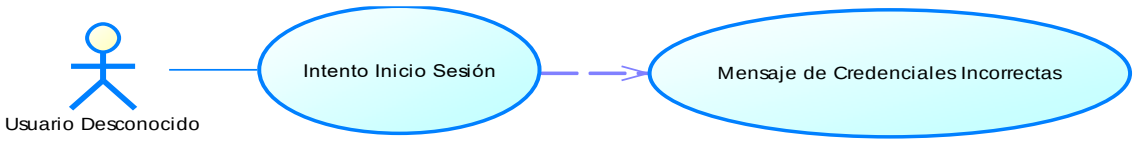
Caso de uso Ingresar a la aplicación

Descripción	**Ingresar a la aplicación**
Objetivo	Permite al usuario propietario poder iniciar sesión
Acciones básicas	• Cerrar Sesión • Cambiar Tema de Aplicación • Visualizar contenido
Post-Condición	La aplicación permite ajustar el brillo y cambiar la contraseña

Nota. La tabla muestra información detallada del caso de uso de ingresar a la aplicación

Figura 13

Diagrama de caso de uso de intento fallido de ingreso



Nota. El grafico muestra el caso de uso del intento fallido de ingreso

Tabla 2

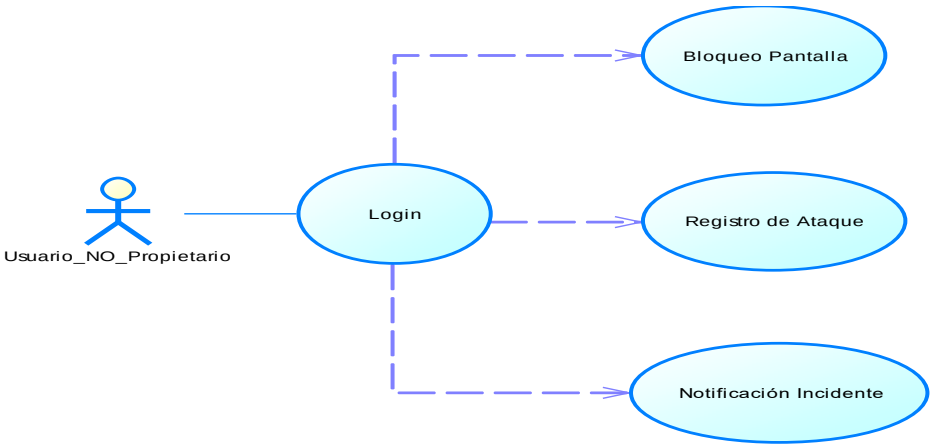
Caso de uso Intento fallido de ingreso

Descripción	**Errar ingreso a la aplicación**
Objetivo	Deniega al usuario desconocido poder ingresar a la aplicación
Acciones básicas	• Mensaje de credenciales incorrectas
Post-Condición	El sistema hace un proceso en bucle hasta el próximo ingreso del usuario propietario

Nota. La tabla muestra información detallada del caso de uso de errar el ingreso a la aplicación

Figura 14

Diagrama de caso de uso de ingreso de usuario no propietario



Nota. El grafico muestra el caso de uso de ingreso de usuario no propietario

Tabla 3

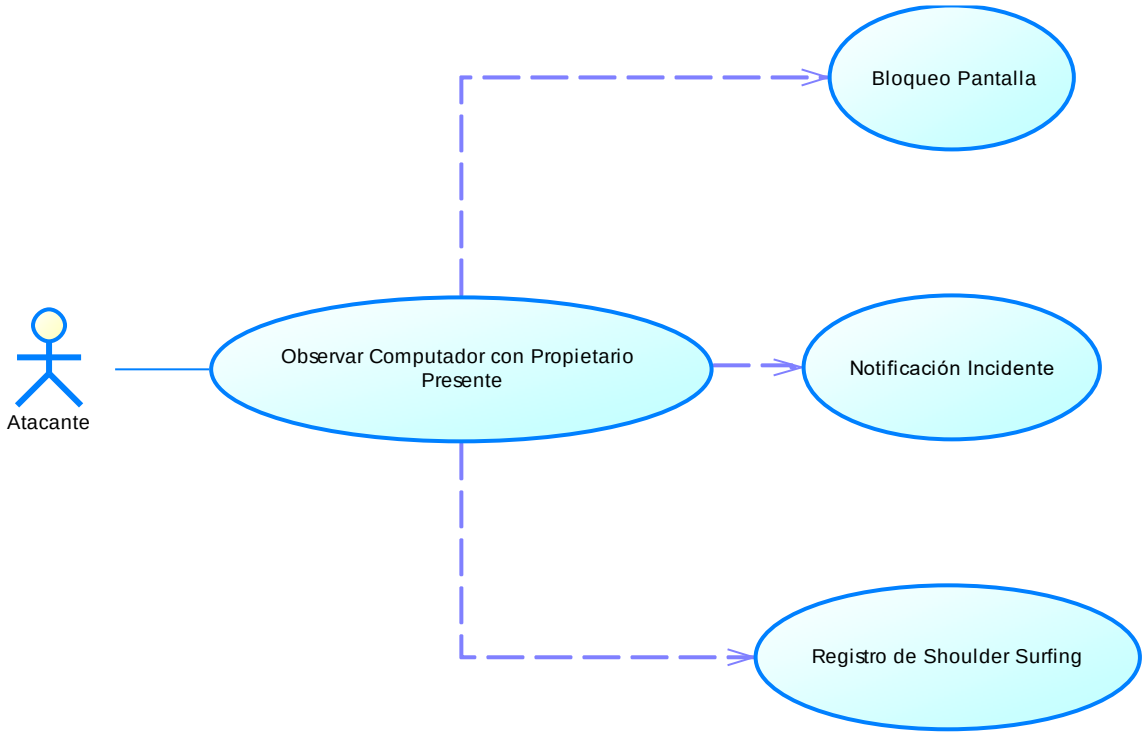
Caso de uso Ingreso de usuario no propietario

Descripción	**Ingresar a la aplicación con usuario no propietario**
Objetivo	Bloquea el acceso a la aplicación al usuario no propietario y notifica al usuario propietario
Acciones básicas	• Bloqueo de Pantalla • Registro de Ataque • Notificación Incidente
Post-Condición	El usuario propietario es alertado de una acción maliciosa

Nota. La tabla muestra información detallada del caso de uso de ingreso de usuario no propietario a la aplicación

Figura 15

Diagrama de caso de uso de actividad de Shoulder Surfing



Nota. El grafico muestra el caso de uso de observar el computador con el propietario presente

Tabla 4

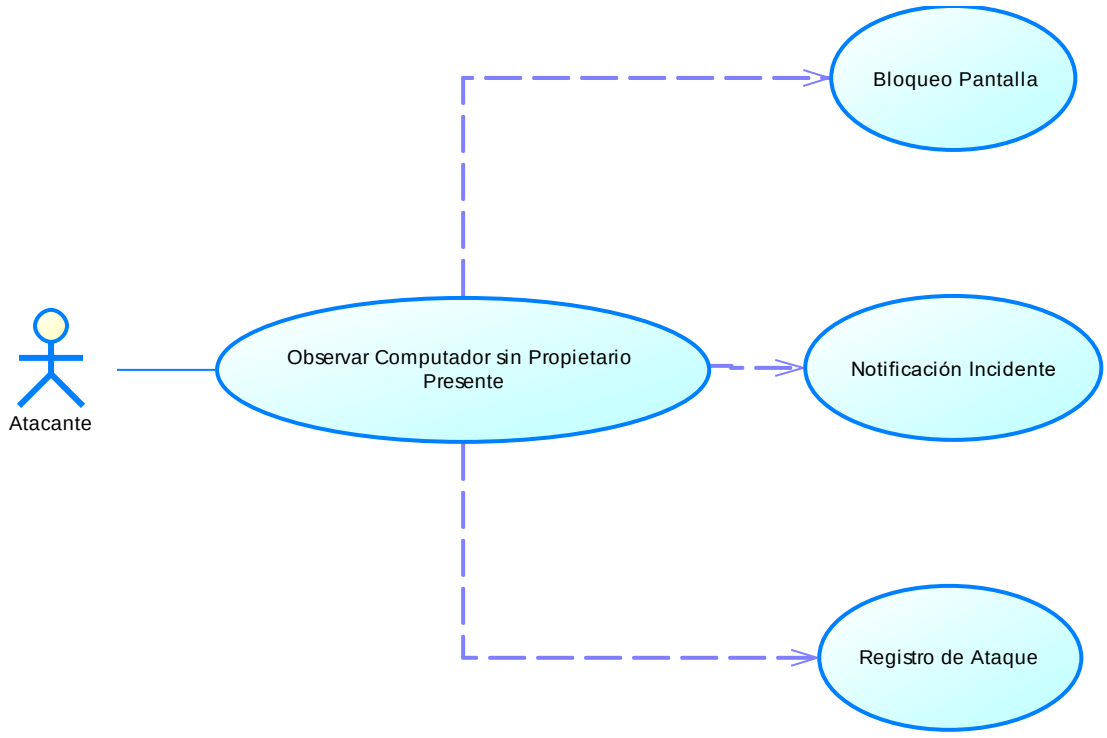
Caso de uso Actividad de Shoulder Surfing

Descripción	**Actividad de Shoulder Surfing**
Objetivo	Bloquea la pantalla del dispositivo si detecta Shoulder Surfing y notifica al usuario propietario para que revise su entorno
Acciones básicas	• Bloqueo de Pantalla • Registro de Shoulder Surfing • Notificación Incidente
Post-Condición	El usuario propietario es alertado de una acción maliciosa

Nota. La tabla muestra información detallada del caso de uso de actividad de Shoulder Surfing.

Figura 16

Diagrama de caso de uso de actividad de Ataque



Nota. El grafico muestra el caso de uso de observar el computador con el propietario ausente

Tabla 5

Caso de uso Actividad de Ataque

Descripción	**Actividad de Ataque**
Objetivo	Bloquea la pantalla del dispositivo si detecta Ataque y notifica al usuario.
Acciones básicas	• Bloqueo de Pantalla • Registro de Ataque • Notificación Incidente
Post-Condición	El usuario propietario es alertado de una acción maliciosa
Descripción	Actividad de Ataque

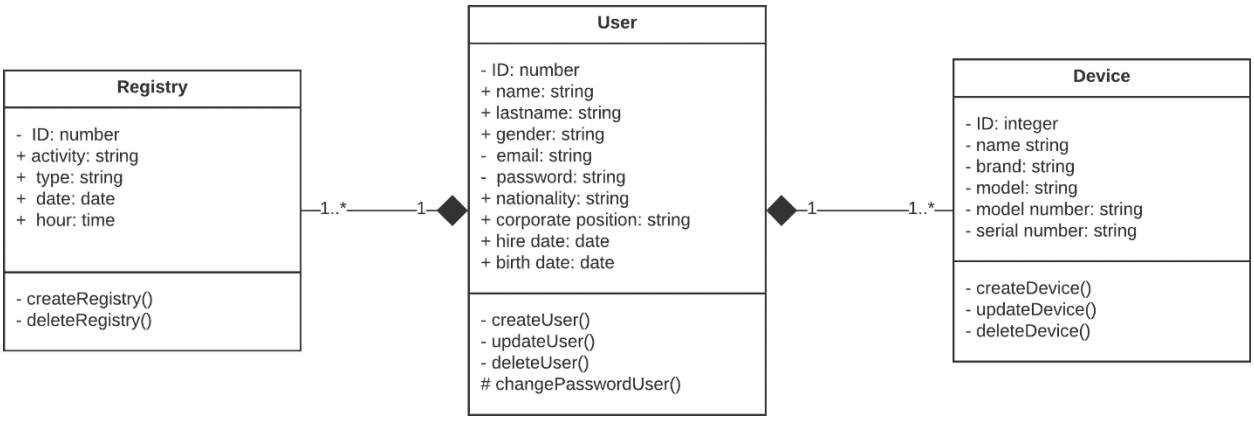
Nota. La tabla muestra información detallada del caso de uso de actividad de Ataque.

Diagrama de Clases

El diagrama de clases brinda una perspectiva general de la estructura de los datos, sus clases y sus respectivas relaciones.

Figura 17

Diagrama de clases de la Base de Datos



Nota. El grafico muestra la representación detallada del diagrama de clases

Gestor de Bases de Datos - Oracle

Para poder gestionar todos los datos que se recolectan, se necesita de una herramienta que pueda gestionar la información, a esta se la conoce como Sistema Gestor de Base de Datos o SGBD. Esta herramienta brinda la facilidad de crear, gestionar y administrar las bases de datos. Existen varios SGBD como MySQL, MariaDB, PostgreSQL, SQLServer, entre otros. No obstante en este caso se optó por Oracle, ya que brinda una excelente estabilidad y escalabilidad, es capaz de trabajar en varias plataformas, y tiene un excelente desempeño en las transacciones de datos («Los gestores de bases de datos (SGBD) más usados», 2019).

Oracle además ofrece una amplia variedad de servicios, que lo ha catalogado como uno de los proveedores de gestión de información que están en la cúspide del mercado, lo que ha incrementado su prestigio y la confiabilidad en corporaciones de menor a gran escala. Su compromiso como empresa es un crecimiento constante, que durante el proceso se fortalezca sus consolidaciones y evolucione a la vanguardia de la tecnología (*ORACLE Base de datos*, s. f.)

Existen varias versiones de Oracle, no obstante, por características de rendimiento, flexibilidad y disponibilidad se ha optado por la versión 11 Express Edition. Es una base de datos gratuita que se puede acoplar a cualquier necesidad de desarrollo para manejar data y tiene los recursos necesarios para aplicaciones en proceso de desarrollo, para las etapas de prueba y de inicio de producción (*Oracle Database*, 2022).

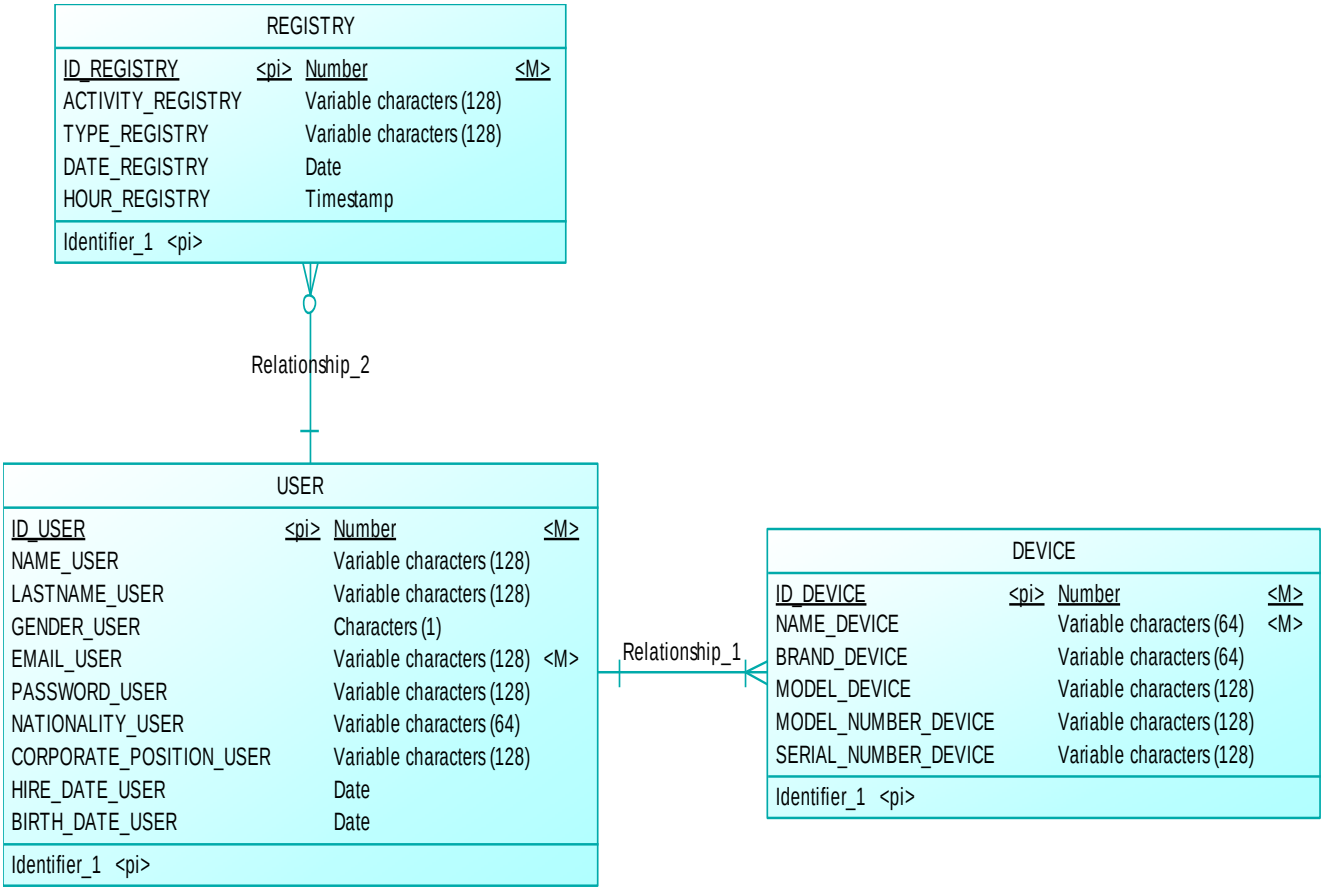
Durante el desarrollo de la aplicación se ha configurado la cadena de conexión a un servidor de base de datos, esto para poder almacenar la información de usuario, dispositivo de usuario y de registro de incidencias de usuario, que de forma más detallada se puede observar en la Figura 18**,** en donde se encuentra la parte del diseño lógico de la base de datos, que representa las diferentes tablas y de que manea se relacionan.

Diseño Lógico de la Base de Datos

Consecuencia del Diagrama de Clases, con una información más detallada y precisa de los tipos de datos y sus longitudes.

Figura 18

Diseño Lógico de la Base de Datos



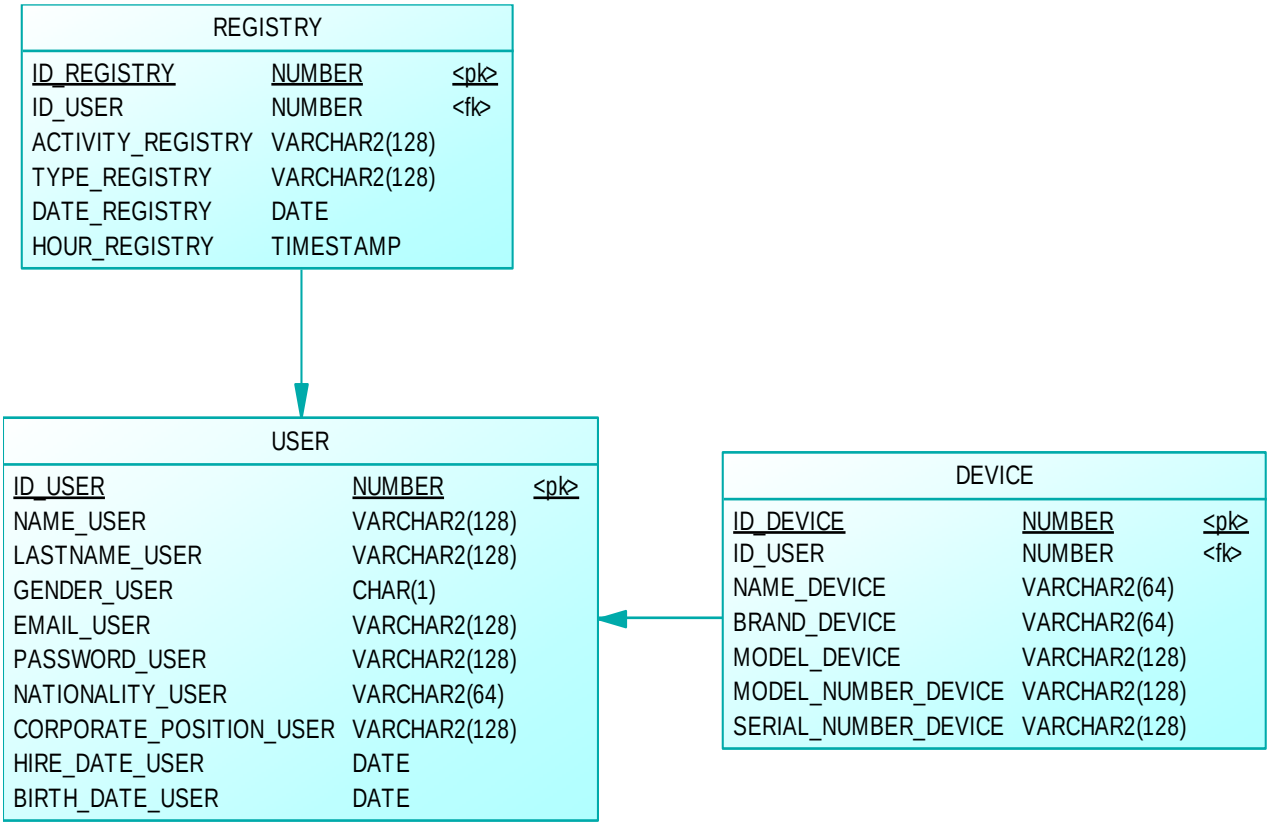
Nota. El grafico muestra la representación detallada del diseño lógico de la base de datos

Diseño Físico de la Base de Datos

El diseño físico de la base de datos, muestra a detalle la representación de la claves primarias y foráneas que están en cada tabla.

Figura 19

Diseño Físico de la Base de Datos



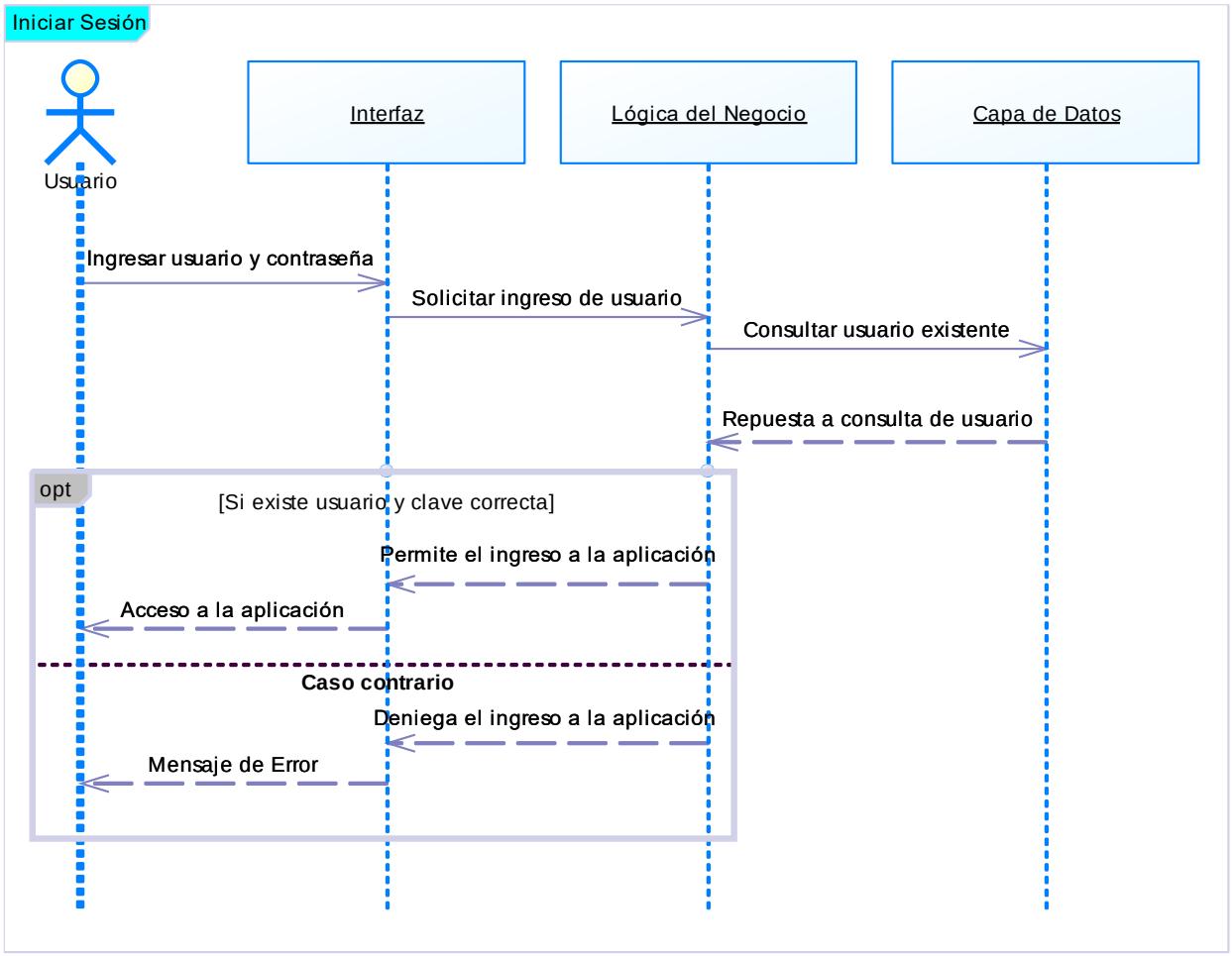
Nota. El grafico muestra la representación detallada del diseño físico de la base de datos

Diagramas de Secuencia

Los diagramas de secuencia permiten esquematizar las interacciones que se realiza la aplicación entre sus diferentes procesos, modelos y actores. En la Figura 20, Figura 21 y Figura 22 se puede observar los diagramas de secuencia para el inicio de sesión, la autenticación facial y el cambio de contraseña.

Figura 20

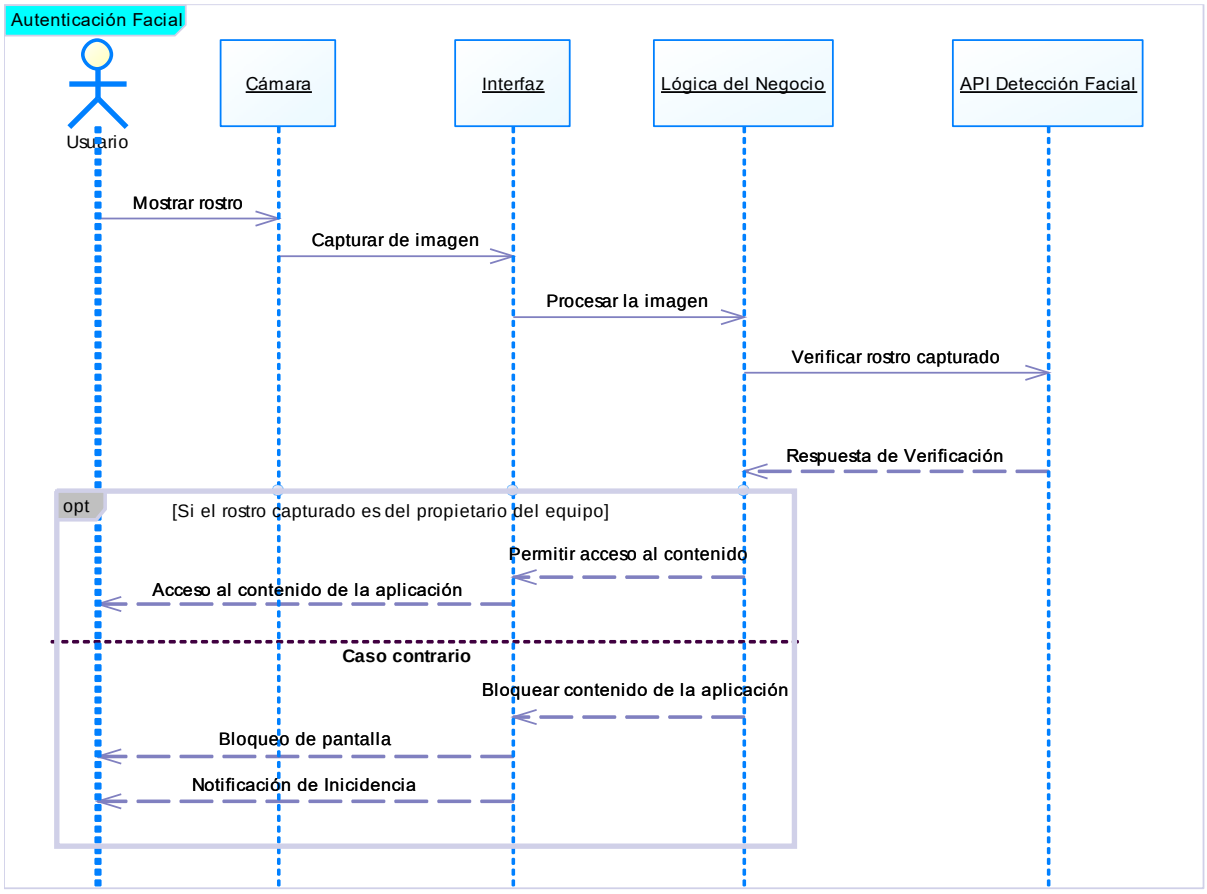
Diagrama de secuencia de Iniciar Sesión



Nota. El grafico muestra la secuencia de inicio de sesión

Figura 21

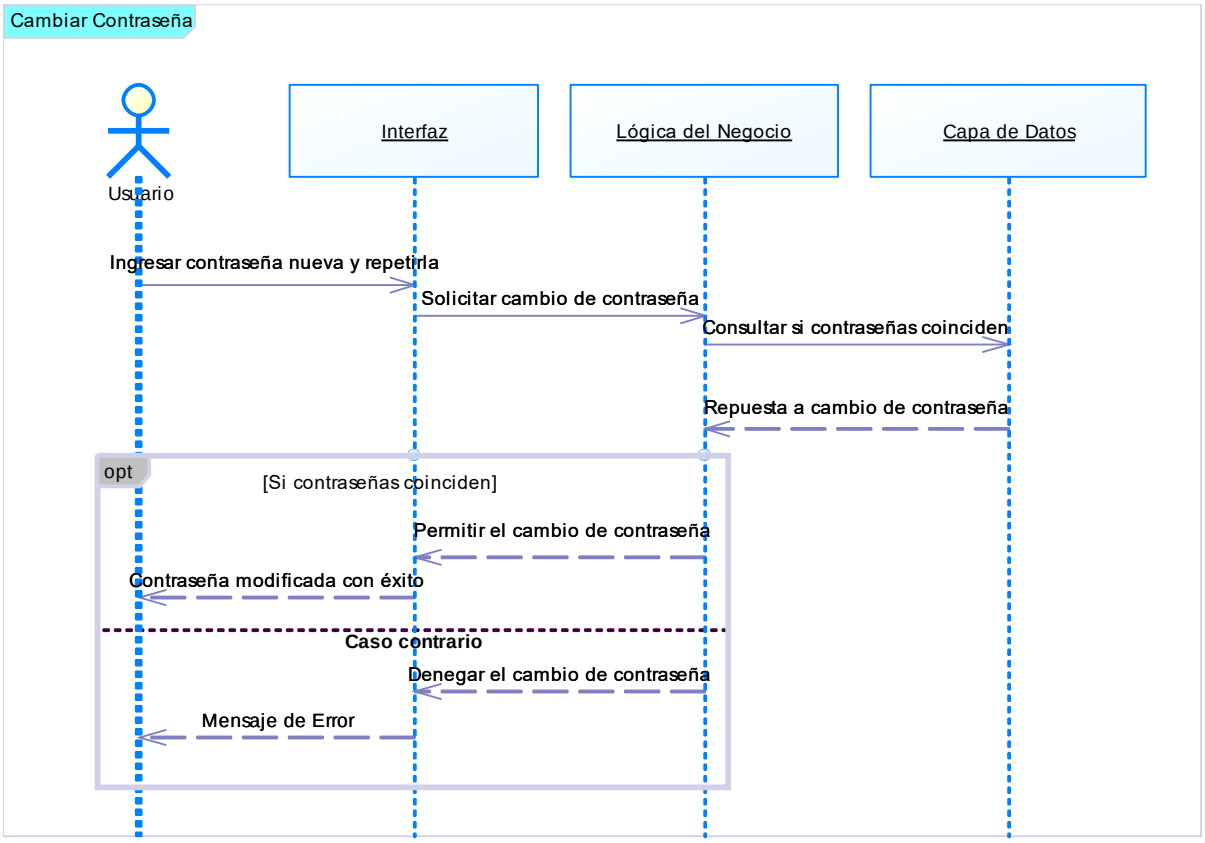
Diagrama de secuencia de Autenticación Facial



Nota. El grafico muestra la secuencia de autenticación facial.

Figura 22

Diagrama de secuencia de Cambiar Contraseña



Nota. El grafico muestra la secuencia de cambio de contraseña

Fase de Desarrollo

Metodología de Programación

Es importante definir una metodología de programación cuando se empieza un desarrollo de software por el hecho de coordinar a los colaboradores del equipo de trabajo. Esto sirve a los desarrolladores para tener un lineamiento de trabajo, saber que deben hacer y hasta cuando lo pueden hacer. En este proyecto se ha optado por usar SCRUM que es una metodología ágil que se centra en la unión, adaptabilidad y en la entrega de tareas por sprints, estos son ciclos reiterativos constantes de tiempo corto que se planifican previo el inicio del trabajo, en donde cada desarrollador debe cumplir distintas tareas de creación, corrección o

mejora (Baxter et al., 2022) Además, en SCRUM es primordial la comunicación entre los miembros de trabajo, lo que provoca una colaboración de código y de ideas para solventar soluciones. Lo que resalta es que esta metodología se focaliza en llegar al objetivo final, esto por la entrega continua de resultados que desemboca en requerimientos finiquitados o en retroalimentaciones para su posterior mejora o corrección (*Why Does Scrum Work?*, s. f.).

Python, aplicaciones de escritorio y Machine Learning

A medida que pasan los años cada vez emergen lenguajes de programación que llegan a tener una gran popularidad y demanda, uno entre ellos es Python un lenguaje de programación que ha llegado a desbancar a lenguajes como Java, C y C++ abriéndose su propio paso en la cima según lo menciona (Saabith et al., 2023) en su estudio, en donde a su vez también señala que Python ha llegado a eso gracias a las características que se indican a continuación en la Tabla 6.

Tabla 6

Características y definiciones de Python

Características	Definiciones
Simple y Fácil de aprender	Es fácil de leer y aprender, debido a que es muy parecido en el idioma inglés.
Comunidad de apoyo	Tiene documentación oficial como tutoriales de YouTube, a modo que programadores de todas las edades y habilidades encuentren apoyo.
Desarrollo web	Python se utiliza principalmente para el desarrollo web, además tiene una variedad de marcos para el desarrollo de sitios web y frameworks tales como Django, flask, pylon y así sucesivamente.

Características	Definiciones
Uso en big data y machine learning	Es utilizado como una herramienta para el análisis y la ciencia de datos para las organizaciones, además gracias a librerías como OpenCV para visión por ordenador y TensorFlow para crear redes neuronales lo que permite ser usada en miles de proyectos de aprendizaje automático.
Eficacia	Posee un paradigma de programación "pitónica" que permite hacer más con menos trabajo, sin mencionar su versatilidad para desarrollar aplicaciones multiplataforma; web, móvil, aplicaciones de escritorio y programación de hardware.
De alto nivel	Maneja una fuerte abstracción de los detalles de la plataforma subyacente o la máquina, también utiliza elementos del lenguaje natural, de modo que es más fácil de usar y automatizar.
Tipado Dinámico	Las variables, objetos, ente otros componentes se infieren por lo general durante el tiempo de ejecución, por lo que no se pueden asignar o declaran estáticamente.
Portátil, independiente de la plataforma y multiplataforma	Permite realizar aplicaciones portables para Windows, Linux y Mac OS, no obstante, se deberá llevar el debido cuidado con cualquier dependencia específica, además se puede utilizar en servidores y también de pequeños dispositivos como la Raspberry Pi3.
Interpretado	Es interpretado diferenciándose del compilado que requiere que el código se convierta del lenguaje original de máquina. Este no requiere compilación a lenguaje máquina, no obstante, lo convierte a código byte y luego lo traduce al lenguaje nativo de la máquina subyacente.
Multiparadigma	Permite el uso de varios paradigmas de programación e implementación de entre los cuales se puede señalar la clásica programación orientada a objetos o la programación procedimental.

Características	Definiciones
Extensible	Admite y ejecuta fragmentos de código a través de la escritura de esa parte del código en C o C++ y luego es utilizada por medio del programa Python y viceversa otorgándole capacidades de scripting al lenguaje de programación.
Bibliotecas extensas	Incluye módulos incorporados escritos en C para acceso a la funcionalidad del sistema, tales como operaciones de E/S, así como también módulos escritos en Python.
Recogida de basura	Podemos olvidarnos de gestionar la memoria, no obstante, de necesitarse Python tiene la interfaz Garbage Collector la cual administra la recolección de basura.

Nota. La tabla muestra las características de Python que hacen sobresalir a este lenguaje de programación de entre los demás, esto permite señalar el porqué de su importancia y preferencia por los desarrolladores.

De entre todas las características que se mencionan en la Tabla 6 aquellas correspondientes al uso en Big Data y Machine Learning, eficacia, portátil, independiente de la plataforma y multiplataforma, bibliotecas extensas, recogida de basura, son las cuales sin duda alguna son fundamentales para la elaboración de la aplicación propuesta, ya que cada de las mencionadas proporcionan un camino seguro a la realización de la aplicación que se plantea realizar a través del uso de tecnologías como el reconocimiento facial, librerías para el control de algunas características y funcionalidades del sistema operativo, la eficacia con respecto a los procesos que se manejan versus el consumo de memoria y recolección de basura, la portabilidad e independencia de la plataforma la cual no restringe o limita el desarrollo de aplicaciones. Cabe destacar que este lenguaje de programación guarda una adecuada interacción con base de datos y protocolos de internet tales como HTTP, SMTP, XML-RPC, FTP y POP por consiguiente se lo ha llevado a utilizar en sectores aeroespaciales, bancarias, aplicaciones empresariales de tal forma que gigantes de la tecnología como Cisco, IBM,

Mozilla, Google, Quora, Hewlett-Packard, Dropbox, y Qualcomm utilizan este lenguaje (Saabith et al., 2023).

Con todo lo anteriormente visto en mente, además de las características no mencionadas explícitamente ayudan a establecer este lenguaje de programación a ser el predilecto para este desarrollo, todo esto es corroborado por (Saabith et al., 2023) donde adicionalmente señala que gracias a su sintaxis sencilla, una arquitectura modular herramientas de procesamiento de texto y la capacidad de trabajar en múltiples sistemas operativos que lo convierten en una opción deseable para desarrollar aplicaciones de escritorio en las cuales se pueda crear interfaces gráficas de usuario (GUI) altamente funcionales.

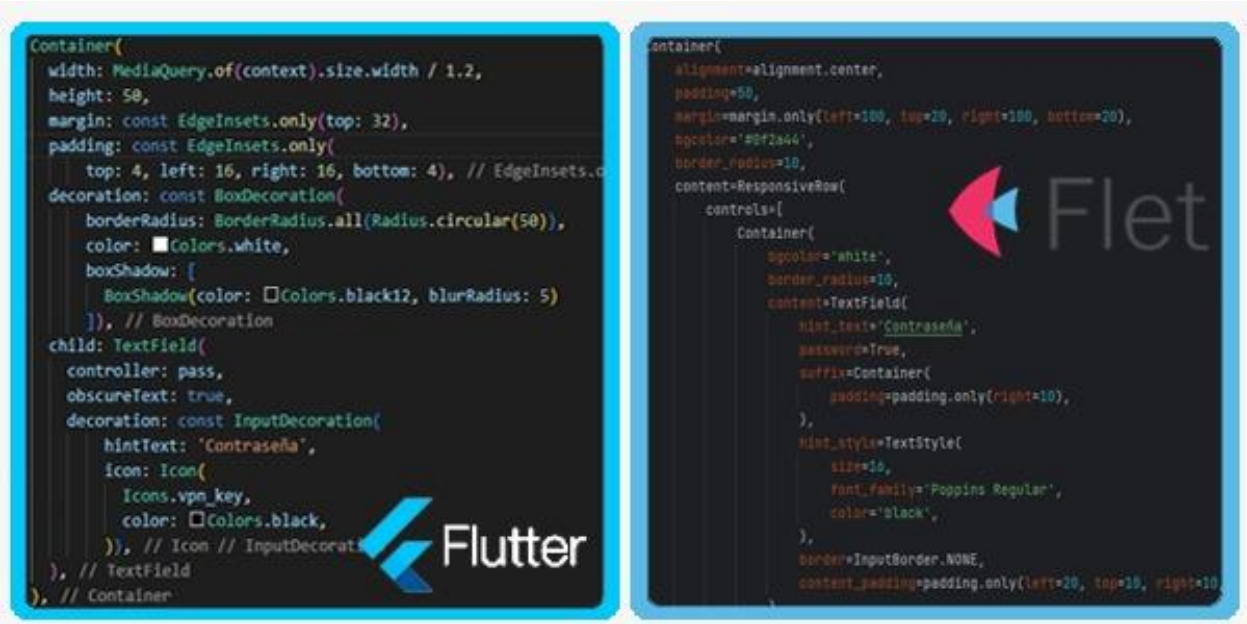
Flet

Una de las tareas un tanto complejas una vez seleccionado el lenguaje de programación base para el desarrollo de la aplicación es encontrar una librería para crear interfaces gráficas, la cual cumpla con los requerimientos de compatibilidad, multiplataforma y portabilidad, resulta una tarea de lo más importante, por lo que tras una investigación en repositorios de documentación de GitHub, tutoriales de YouTube, blogs y páginas de respuestas a problemas propuestos por los desarrolladores como stackoverflow, las librerías finalistas fueron DearPyGUI, Flet, Kivy, Tkinter y PyQt5, el paso siguiente fue seleccionar cual es la más sencilla con respecto a sintaxis y facilidad de aprendizaje, esto por lo general no sería un punto tan crítico para la selección de una librería para crear interfaces gráficas, sin embargo, la disponibilidad de tiempo para la realización del aplicativo fue de aproximadamente de cuatro meses, lo que conllevo a tener este factor en consideración, por lo cual el aspecto que se tomó como discriminante a seleccionar una única librería de GUI entre las ya mencionadas fue la relación/similitud con otros lenguajes de programación utilizados o aprendidos durante la formación académica previa a la obtención del título de “Ingeniero en Tecnologías de la Información”, se toma en cuenta que la misma deberá cumplir con todos los

requerimientos necesarios que se nombraron con anterioridad, lo que llevo consecuentemente a definir a Flet como la librería de GUI establecida para este desarrollo, gracias a su relación/similitud con Flutter como se puede ver en la Figura 23. Flutter es un framework de código fuente abierto de desarrollo de aplicaciones móviles creado por Google (*Flutter - Crea Hermosas Aplicaciones Nativas En Tiempo Récord*, s. f.), este framework fue aprendido en el transcurso del aprendizaje en las materias correspondientes al desarrollo de aplicaciones de la malla curricular actual en las cuales se aprendió a desarrollar aplicaciones en Android y Web.

Figura 23

Flutter y Flet comparativa



Nota. La figura muestra fragmentos de código en los cuales se puede ver ciertas similitudes en relación a la sintaxis y componentes entre ambos frameworks.

Ahora que se tiene un claro ganador se comentará más a detalle varios aspectos de este framework tales como; no requiere para su uso experiencia en frontend, adicionalmente permite a los desarrolladores crear fácilmente aplicaciones web, móviles y de escritorio interactivas de gran aspecto en tiempo real en Python donde sólo necesitas tu IDE o editor de

texto favorito sin la necesidad de preocuparse de SDKs, miles de dependencias y herramientas complejas. Tiene una arquitectura sencilla que facilita la vida a los desarrolladores lo que permite obviar el uso de complejas arquitecturas con JavaScript frontend, REST API backend, base de datos, caché, entre otras. Con Flet sólo se necesita escribir un monolito statefull app y obtener Single-Page Application (SPA) multiusuario y en tiempo real (*Flet*, 2022/2023).

Como se mencionó con anterioridad Flet UI está construido con Flutter es por ello que se puede ver algunas similitudes en su código para ello se puede ver Figura 23 en donde claramente se puede ver similitudes en sintaxis, no obstante a pesar de existir estas similitudes, Flet simplifica el modelo de Flutter a través de la combinación de "widgets" de menor tamaño a "controles" listos para ser usados a través de un modelo de programación imperativo a aparte usa un lenguaje agnóstico a modo de que cualquier persona con otro lenguaje de programación podría desarrollar aplicaciones. Finalmente, es posible desplegar la app Flet como una app web y visualízate en un navegador o empaquetarla como una app de escritorio independiente para Windows, macOS y Linux, también es posible instalarla en móviles como una aplicación web progresiva PWA o visualizarla a través de la app Flet para iOS y Android (*Flet*, 2022/2023).

Visión por computador (Open CV) e interacción con la aplicación

La librería Open CV como se señala en varios estudios como por ejemplo en (Hussain & Balushi, 2020), (Khan et al., 2019), (Shreya, 2021), (Zhu & Cheng, 2020) representa un apoyo indiscutible debido a las ventajas en la programación de visión por computador con Python. En (Hussain & Balushi, 2020) se hace referencia sobre complicaciones con respecto a la visión por ordenador, no obstante cuando llegaron las tecnologías modernas aportaron de manera favorable a la resolución de este tipo de problemas que fueron derivados de las condiciones naturales del ser humano tales como el cambio de edad, el uso de accesorios y las variaciones de luz. Para (Khan et al., 2019) se puede encontrar menciones de esta librería de código abierto de Intel “Open CV” en las cuales se habla de que la implementación de esta librería

permite que el reconocimiento facial puede realizarse de una forma rápida y fiable en donde además se indica que puede ofrecer una colección de funciones avanzadas como la detección facial, el seguimiento facial, el reconocimiento facial y una serie de métodos listos para ser usadas en conjunto con desarrollo para la inteligencia artificial (IA) dentro de un marco multiplataforma que apoya a Windows y Macos, así como Mac OS X recientemente. El estudio de (Shreya, 2021) también se llega a mencionar sobre Open CV como una librería que integra formas rápidas de calcular ecuaciones y aplicarlas a píxeles en el manejo de imágenes en especial imágenes que se encuentren en un formato de escala de grises BW, esto da como resultado una matriz bidimensional y explica la relación que existe entre imágenes, imágenes RGB, píxeles y cómo es posible reconstruir una imagen en formato RGB a valores de escala de grises a través de esta librería. Por último y no menos importante se tiene a (Zhu & Cheng, 2020) en donde se propone un algoritmo de seguimiento de actitud eficiente para el reconocimiento facial basado en Open CV el cual utiliza visión por computador, esto permite a los desarrolladores acceder a rutinas utilizadas para aplicaciones de visión por ordenador en la API (Application Program Interface).

Después de haber visto algunos estudios la implementación de Open CV de una forma u otra para ser un complemento a la inteligencia artificial o directamente a el reconocimiento facial se determinó que lo más factible sería el implementar esta librería para evitarnos posibles contratiempos. Conforme se desarrolló esta aplicación en concreto el módulo que tiene la funcionalidad o trabajo en conjunto de realizar el reconocimiento facial, utiliza esta librería en donde las principales funciones que fueron de utilidad son las que se mencionan a continuación en la Tabla 7 que da información sobre la función utilizada y la Tabla 8 señala la utilidad en la aplicación y la línea de código que se realizó con esa función.

Tabla 7

Documentación de funciones utilizadas de Open CV en la aplicación

Módulo	Función	Descripción	Parámetros
videoio. Video I/O	cv2.VideoCapture (int index, int apiPreference = CAP_ANY)	Permite abrir conexión con una cámara para capturar vídeo.	**index** - id del dispositivo de video captura. El valor de 0 para abrir la cámara predeterminada. **apiPreference** - es el backend del API de captura se puede implementar un lector específico, por ejemplo, cv::CAP_DSHOW o cv::CAP_MSMF o cv::CAP_V4L.
	cv2.VideoCapture.isOpened ( )	Indica true si ya se ha inicializado la captura de vídeo, caso contrario false.	
	cv2.VideoCapture.read ( )	Realiza la captura, decodificación y devuelve fotogramas de vídeo.	
imgcodecs. Lectura y escritura de archivos de imagen	cv2.imencode (const String & ext, InputArray img, std::vector< uchar > & buf, const std::vector< int > & params = std::vector< int >())	Almacena en memoria la imagen codificada en un bufer que se redimensiona para ajustarse al resultado.	**ext** – define formato del archivo de salida. **img** - imagen que se va a escribir. **buf** - búfer de salida redimensionado. **params** - parámetros adicionales específicos del formato.
imgproc. Procesamiento de imágenes	cv2.resize (InputArray src, OutputArray dst, Size dsize, double fx = 0, double fy = 0, int interpolation = INTER_LINEAR)	Redimensiona el tamaño de una imagen de modo que el tamaño y el tipo se derivan de src, dsize, fx, y fy.	**src** - imagen de entrada. **dst** - imagen de salida. **dsize** - tamaño de la imagen de salida. **fx** - factor de escala horizontal. **fy** - factor de escala vertical. **interpolation** - método de interpolación.

Módulo	Función	Descripción	Parámetros
imgproc. Procesamiento de imágenes	cv2.rectangle (InputOutputArray img, Point pt1, Point pt2, const Scalar & color, int thickness = 1, int lineType = LINE_8, int shift = 0)	Dibuja un rectángulo según dos esquinas opuestas pt1 y pt2.	**img** - imagen. **pt1** - vértice del rectángulo. **pt2** - vértice del rectángulo opuesto a pt1. **color** - color. **thickness** - grosor de las líneas del rectángulo. **lineType** - tipo de línea. **shift** - bits fraccionarios en las coordenadas del punto.
	cv2.putText (InputOutputArray img, const String & text, Point org, int fontFace, double fontScale, Scalar color, int thickness = 1, int lineType = LINE_8, bool bottomLeftOrigin = false)	Dibuja una cadena de texto en la imagen.	**img** - imagen. **text** - cadena de texto a dibujar. **org** - esquina inferior izquierda. **fontFace** - tipo de fuente. **fontScale** - factor de escala de la fuente. **color** - color. **thickness** -grosor de del texto. **lineType** - tipo de línea **bottomLeftOrigin** – verdadero cuando el origen de los datos de la imagen está en la esquina inferior izquierda caso contrario se sitúa en la esquina superior izquierda.
cudaarithm. Operaciones con matrices	cv2.flip (InputArray src, OutputArray dst, int flipCode, Stream & stream = Stream::Null())	Gira una matriz 2D alrededor de los ejes vertical, horizontal o ambos.	**src** - matriz de origen. **dst** - matriz de destino. **flipCode** - modo de volteo. **stream** - corriente para la versión asíncrona.

Nota. La tabla muestra la correspondiente documentación de las funciones utilizadas para el desarrollo de la aplicación. Tabla creada a partir de la documentación de (*OpenCV: OpenCV modules*, s. f.).

Tabla 8

Uso de funciones de Open CV en la aplicación y su utilidad

Función	Línea de código	Utilidad en la aplicación
cv2.VideoCapture	self.video_cap = cv2.VideoCapture(0)	Crea un objeto de captura de video para extraer las imágenes del dispositivo de video grabación principal por defecto conectado a la computadora.
cv2.VideoCapture.isOpened	while self.video_cap.isOpened():	Utilizado como validación con el fin de verificar la existencia de un dispositivo de video grabación en la computadora o verificar el correcto funcionamiento del mismo.
cv2.VideoCapture.read	ret, image = self.video_cap.read()	Línea de código utilizada para asignar a una variable los videos frames obtenidos por el dispositivo de grabación.
cv2.flip	image = cv2.flip(image, 1)	Fragmento de código utilizado para crear el efecto espejo en el frame de la video grabación y ser almacenado en la variable imagen.

Función	Línea de código	Utilidad en la aplicación
cv2.imencode	_, image_arr = cv2.imencode('.png', frame)	Se codifica una imagen en un búfer de memoria, para posteriormente ser utilizada para convertir la imagen de formato matriz a bytes codificados para obtener salida de video visible en la aplicación.
cv2.resize	small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)	Se cambia el tamaño del frame de vídeo a 1/4 para acelerar el procesamiento del reconocimiento facial.
cv2.rectangle	cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)	Línea de código para resaltar los frames de n rostros detectados en la imagen a través de un recuadro de color rojo sin relleno.
	cv2.rectangle(frame, (left, bottom - 35), (right, bottom), (0, 0, 255), cv2.FILLED)	Se crea un recuadro con relleno del mismo color en la parte inferior del anterior recuadro.
cv2.putText	cv2.putText(frame, name, (left + 6, bottom - 6), cv2.FONT_HERSHEY_DUPLEX, 0.8, (255, 255, 255), 1)	Línea utilizada para situar una cadena de texto de color blanco correspondiente al nombre y el porcentaje de exactitud del reconocimiento de la persona, este será puesto en el recuadro con relleno.

Nota. La tabla muestra las líneas de código utilizadas con las correspondientes funciones de Open CV anteriormente mencionadas, además de su utilidad reflejada en la aplicación.

Face Recognition

Una vez compendio el valioso aporte de Open CV a través de sus módulos y funciones, para generar y modificar data que posteriormente será utilizada en el reconocimiento facial, sin embargo, ¿por qué necesariamente se debería utilizar una herramienta que utilizará Deep Learning y redes neuronales NN? La razón principal aparte de que es una tecnología moderna, es que varias soluciones que se han desarrollado con IA por lo general utilizan Machine Learning, no obstante en la actualidad la evolución de una de las ramas del ML, el Deep Learning ha logrado obtener una gran precisión comparado con los algoritmos de ML tradicionales de tal forma que los modelos DL funcionan mejor que los modelos ML en términos de precisión (Benavides-Astudillo et al., 2023), un factor fundamental para el desarrollo del presente trabajo y aplicación debido a la necesidad de precisión al diferenciar rostros de personas.

Una vez establecida la necesidad de encontrar una librería o algoritmo basado en Deep Learning que a su vez utilice redes neuronales del tipo convolucional CNN, esta librería o algoritmo al mismo tiempo ayudará a cumplir con los objetivos trazados en un inicio para la creación de esta aplicación. A fin de encontrar lo mencionado era necesario realizar una investigación la cual requeriría encontrar documentación y estudios con el fin de hallar el reconocimiento facial ideal para la aplicación. Entre los estudios que se logró encontrar se tiene a (Cárabe & Cermeño, 2021) donde menciona la precisión de algunos de los algoritmos de reconocimiento facial en donde se tiene a deepFace con una precisión del 97.35%, DeepID3 99.43%, FaceNet 99.63%, VGGFace 98.95%, y Arcface 99.83%. En (Li et al., 2020) se menciona sobre otros famosos algoritmos de reconocimiento facial con su correspondiente porcentaje de reconocimiento, los que se indicaron fueron Deep Face 97.35%, FaceNet 99.63%, DeepFR 98.95%, DeepID2+ 99.47%,Center Face 99.28%, Baidu 99.13%,SphereFace 99.42%, VGGFace 99.13%, Face++ 99.50%, FR+FCN 96.45%, DeepID 97.45%, GaussianFace

98.52%,DeepID2 99.15%, DeepID3 99.53%, YouTu Lab Tencent 99.80%, PingAn AI Lab 99.80%, yunshitu 99.87%, Deepmark 99.23%, Camvi 99.87%, Innovative Technology 99.88%, Fisher vector faces 93.03%, CMD+SLBP 92.58%, Simile classifiers 84.72%, DFD 84.02%, LBP PLDA 87.33%, LBP multishot 85.17%. Los antes citados estudios se basaron en la página web de Labeled Faces in the Wild LFW y el documento de (Learned-Miller et al., 2016) las cuales aportan análisis cuantitativo de precisión y márgenes de error, de prácticamente todos los algoritmos de reconocimiento facial creados hasta el momento y a su vez estos han sido clasificados en Resultados no supervisados, Resultados con imágenes restringidas, sin datos externos, Resultados sin restricciones, sin datos externos, Resultados de datos externos sin etiquetas y con restricción de imágenes, Resultados de datos externos sin restricciones ni etiquetas, Resultados de datos externos no restringidos y etiquetados y Rendimiento humano, medido a través de Amazon Mechanical Turk.

Por otro lado, según (*Top 23 Face-Recognition Open-Source Projects (Aug 2023)*, s. f.) y su top de 23 de algoritmos de reconocimiento facial se pudo encontrar algunos algoritmos que se escogieron debido a los valiosos comentarios de la comunidad, variedad de tutoriales o blogs que estos poseen, etc. Estos fueron Face Recognition, insightface, facenet, deepface, facenet-pytorch, CompreFace y retinaface, los cuales cada uno pasaron un proceso de evaluación en donde los puntos a evaluar fueron la facilidad, aprobación por la comunidad, su arquitectura, sus modelos, y su necesidad de información que en este caso serian el número de imágenes que necesita por persona o archivos complementarios, posteriormente a este análisis se determinó que Face Recognition es el ganador, ya que consiguió obtener una nota aceptable en la mayoría de los puntos y sobresaliente en facilidad, aprobación por la comunidad y arquitectura, esto fue apoyado por (Geitgey, 2017/2023) su repositorio de código en GitHub el cual menciona que Face Recognition es el Api de reconocimiento facial más sencilla del mundo para Python uno de los principales puntos a tener en cuenta para el

desarrollo de este aplicativo, esto permite aportar una forma rápida de entender y desarrollar los cimientos del módulo de reconocimiento facial. Esta librería de reconocimiento facial ha sido construida por medio de la implementación del reconocimiento facial de última generación de Dlib (King, 2009) el cual fue creado por medio de Deep Learning uno de los aportes perfecto para escoger a esta herramienta para la detección e identificación de rostros de personas, además se indica que este modelo tiene una precisión del 99.38% con un margen de error de 0.27 como es referenciado en Labeled Faces in the Wild LFW. Esta herramienta es tan sencilla que incluso a través de línea de comandos se puede hacer el reconocimiento facial de una carpeta de imágenes. De forma general el proceso que realiza esta Api, se señala a continuación:

- Encontrar caras en imágenes: encuentra todas las caras que aparecen en una imagen.
- Encontrar y manipular rasgos faciales en imágenes: realiza esto para encontrar la ubicación y el contorno de los ojos, la nariz, la boca y la barbilla de cada persona.
- Identificar caras en fotos: reconoce/identifica quién aparece en cada foto.

Relación de Face Recognition con Dlib y su interacción con la aplicación

Sobre el Api Face Recognition hay un tema que no debe pasar por alto el cual tiene que ver con Dlib, sin embargo, ¿qué es Dlib? Según (*dlib C++ Library*, s. f.) se describe a Dlib como un moderno conjunto de herramientas desarrolladas en C++ las cuales trabajan bajo una licencia de código abierto lo que permite su utilización gratuita. Dlib busca resolver problemas del mundo real a través de la ejecución de sus características principales, las cuales son documentación, código portable de alta calidad, algoritmos de aprendizaje automático, algoritmos numéricos, algoritmos de inferencia de modelos gráficos, procesamiento de imágenes, threading, redes, interfaces gráficas de usuario, algoritmos de compresión e integridad de datos, pruebas y utilidades generales. Entre todas las características principales mencionadas aquella que más se requiere analizar en este caso es la categoría de algoritmos

de aprendizaje automático, ya que dentro de la misma se puede encontrar el basto mundo del Deep Learning, el cual para Dlib se encuentra una técnica llamada aprendizaje profundo métrico o también conocido como Deep Metric Learning el cual se difiere del convencional entrenamiento de red neuronal que realiza los pasos de i) Aceptar una única imagen de entrada y ii) Emitir una clasificación/etiqueta para esa imagen. Esta técnica difiere en el hecho de que en lugar de obtener una única etiqueta se obtiene un vector de características de valor real también conocido como embedding, el cual tiene 128 números como salida, este vector es utilizado para cuantificar la cara, no obstante para realizar esto lo realiza mediante un entrenamiento denominado "paso de entrenamiento de tripleta" el cual se basa en 3 imágenes faciales en donde 2 de las 3 son de la misma persona de tal modo que el vector de 128 para cada imagen se ajuste los pesos con el objetivo de que los 2 vectores de las 2 imágenes similares se aproximen más que el vector sobrante lo que consecuentemente permite aportar una mejor precisión (Rosebrock, 2018).

La arquitectura de red de Face Recognition para el reconocimiento facial se basa en ResNet-34, está es una red neuronal residual la cual es considerada como una mejora de las CNN en la cual se menciona que se han implementado 29 capas convolucionales. Esta red neuronal se entrenó desde cero con un conjunto de datos de aproximadamente 3 millones de rostros tomados de conjuntos de datos tales como Face Scrub, el conjunto de datos VGG e imágenes que el propio creador ha sacado del Internet (King, s. f.). La razón por la cual se dio a conocer Dlib es debido a que esta fue la base para la creación de Face Recognition, de tal manera que si se explica Dlib se explicaría también Face Recognition prácticamente.

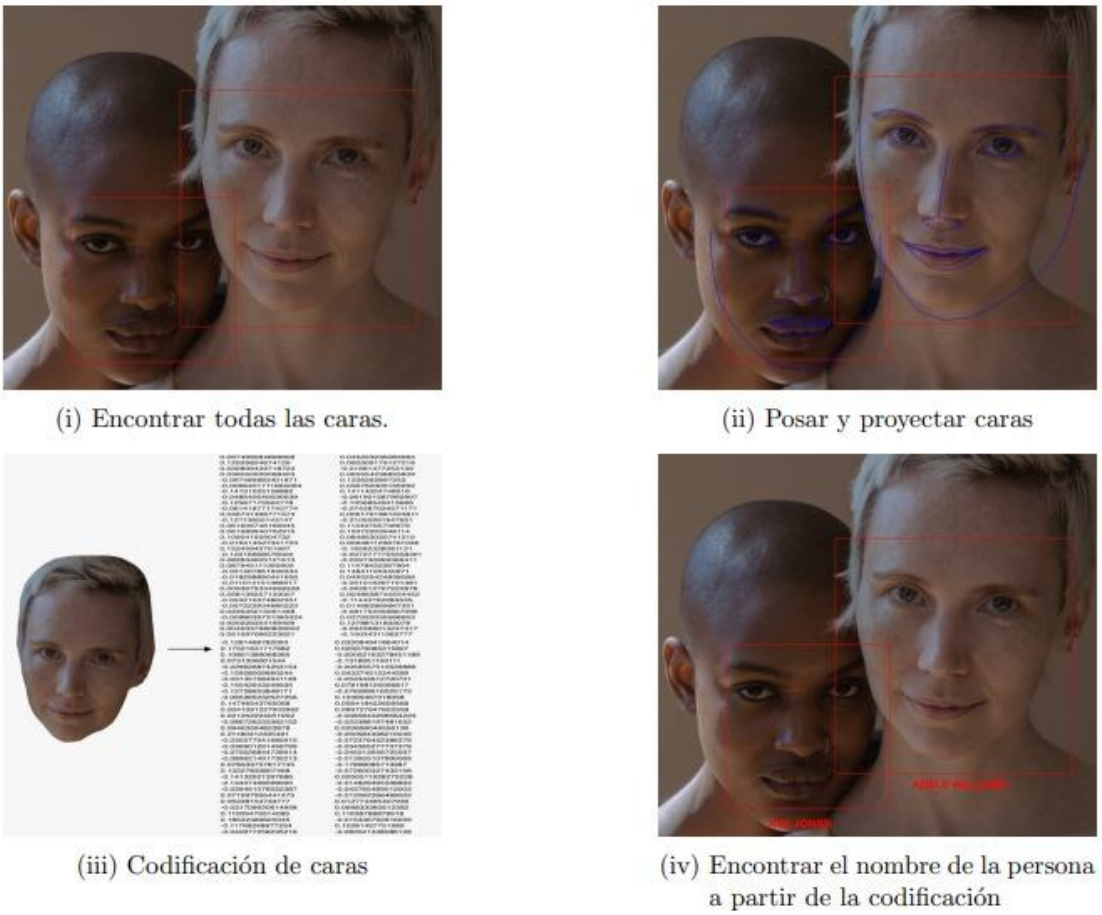
El proceso que llevaron a cabo los creadores para desarrollar este reconocimiento facial se lo puede ver de forma resumida en la Figura 24, de forma detallada el proceso se divide específicamente en 4 pasos los cuales son: i) Encontrar todas las caras, ii) Posar y proyectar caras, iii) Codificación de caras y iv) Encontrar el nombre de la persona a partir de la

codificación. En el primer paso se puede señalar que debe existir una transformación de la imagen a blanco y negro debido a dos razones, la primera porque no necesitamos datos de color para encontrar las caras y la segunda con el objetivo de optimizar el procesos ya que lógicamente pasar de trabajar con 3 matrices a 1 marca la diferencia, por otro lado se debe averiguar la dirección en la que se oscurecen los pixeles; es decir la relación en la que se oscurece un pixel con respecto de sus pixeles próximos, esta dirección tomada en forma de flechas es denominada gradiente y la razón por la que se realiza este trabajo es la de evitar problemas con el análisis de imágenes muy oscuras o muy claras de la misma persona, ya que estas tendrán valores de píxel totalmente diferentes, después del análisis del gradiente, tanto las imágenes muy oscuras como las muy claras tendrán exactamente la misma representación. El segundo paso se consiste en solventar el problema de que una persona puede girar su cara en distintas direcciones, por lo que el reconocimiento facial al igual que el ser humano debería poder identificar a la persona, eso puede solventar mediante algoritmos de estimación de puntos de referencia de la cara, en este caso en concreto el modelo de 68 puntos de referencia que se encuentran distribuidos al contorno de la cara, en la barbilla, el borde exterior de cada ojo, el borde interior de cada ceja y el borde de la nariz, esto permitirá identificar las ubicaciones de las partes del rostro, para posteriormente aplicar transformaciones básicas donde simplemente se giran, se escalan y se recortan las imágenes para que los ojos y la boca estén centrados de manera óptima. Para el tercer paso se busca una forma de codificar el rostro para que después resulte fácil identificarlo, para esto se utilizara la técnica anteriormente descrita del embedding a través del uso de las redes neuronales por medio del Deep Learning, no obstante, lo más importante de esto es que este proceso puede generar embeddings incluso de rostros de personas que no han sido previamente entrenadas. Por último, el cuarto paso que cierra este proceso entran en juego los algoritmos de aprendizaje automático donde en particular para este reconocimiento facial utilizaron un clasificador lineal simple SVM el cual requirió que se entrenase para que pueda tomar las medidas de una nueva imagen de prueba y

decir qué persona conocida es la que más se aproximaba, esto se puede explicar y entender de una forma sencilla como una función y = f(x) la cual desconocemos, no obstante un computador al repetir millones de veces este proceso encuentra los pesos necesarios de f(x) para encontrar y la cual representa a la identificación de la persona y x representa la imagen que es ingresada posteriormente a la generación del modelo, esto resulta tan sencillo como y = mx+b donde m y b serian pesos encontrados por el computador, de tal modo que si se supone que x = 7, m = 0.35 y b = 2 se tendría como resultado un y = 4.45 (Geitgey, 2020).

Figura 24

Pasos necesarios para lograr el Reconocimiento Facial



Nota. El grafico muestra el proceso paso a paso que se requiere para lograr detectar y reconocer rostros de personas.

Una vez entendido todo este proceso lo que resta es implementar estos pasos mediante el Api de Face Recognition, esto se puede ver en los siguientes fragmentos de código tomados del módulo de reconocimiento facial creado, para una descripción y detalle claro se presentan las siguientes tablas Tabla 9 y Tabla 10. En la primera gracias a (*Welcome to Face Recognition's documentation! — Face Recognition 1.4.0 documentation*, s. f.) proporciona la documentación necesaria para que se pueda comprender de mejor manera el funcionamiento general de las funciones del Api de Face Recognition utilizadas, mientras que para la segunda se presenta de forma resumida como es la interacción e implementación en las líneas de código de la aplicación y las funciones mencionadas.

Tabla 9

Documentación de funciones utilizadas de Face Recognition en la aplicación

Función	Descripción	Parámetros
face_recognition.api.load_image_file(file, mode='RGB')	Carga una imagen del tipo .jpg, .png, etc., en un array numpy.	**file** - nombre de imagen a cargar. **mode** - formato al que convertir la imagen. Sólo se admite RGB.
face_recognition.api.face_encodings(face_image, known_face_locations=None, num_jitters=1, model='small')	Retorna una codificación facial de 128 dimensiones para cada rostro de la imagen.	**face_image** - imagen con rostros. **known_face_locations** - opcional - las cajas delimitadoras de cada rostro si ya las conoces. **num_jitters** – número de muestreo, entre más alto, más preciso, sin embargo, más lento. **model** - opcional - qué modelo se utilizará "grande" o "pequeño".
face_recognition.api.face_locations(img, number_of_times_to_upsample=1, model='hog')	Devuelve una matriz de cajas delimitadoras de todos los rostros humanos en una imagen	**img** - una imagen. **number_of_times_to_upsample** – número de muestreos en busca de rostros entre más altos encuentran rostros más pequeños. **model** – selección de modelo "hog" o "cnn" según la CPU o la GPU.

Función	Descripción	Parámetros
face_recognition.api.compare_faces(known_face_encodings, face_encoding_to_check, tolerance=0.6)	Compara codificaciones de un rostro con otro para ver si coinciden.	**known_face_encodings** - lista de codificaciones de rostros entrenados. **face_encoding_to_check** – codificación única de rostros a comparar. **tolerance** - distancia entre los rostros para ver similitudes, cuan más baja mejor.
face_recognition.api.face_distance(face_encodings, face_to_compare)	Compara codificaciones de rostros, con rostros conocidos y obtiene una distancia euclídea que señala la similitud entre ellos.	**face_encodings** - lista de codificaciones de rostros a comparar **face_to_compare** - codificación de rostros con las que comparar.

Nota. La tabla muestra la documentación de las funciones utilizadas en la aplicación desarrollada. Tabla creada a partir de la documentación de (*Welcome to Face Recognition's documentation! — Face Recognition 1.4.0 documentation*, s. f.).

Tabla 10

Uso de funciones de Face Recognition en la aplicación y su utilidad

Función	Líneas de código	Utilidad en la aplicación
face_recognition.api.face_locations	face_image = face_recognition.load_image_file(f"src/features/face_recognition/faces/{image}")	Se guarda en una variable la extracción del contenido de la imagen como matriz del tipo numpy.
face_recognition.api.face_encodings	face_encoding = face_recognition.face_encodings(face_image)[0]	Línea utilizada para generar las listas de codificaciones de caras de 128 dimensiones para cada una de las imágenes nuevas de rostros, se le agrega [0] para que identifique el rostro único de dicha imagen y todo esto almacenarla en una variable.
face_recognition.api.face_locations	self.face_locations = face_recognition.face_locations(rgb_small_frame, model="hog")	Se guarda una lista de tuplas de ubicaciones de caras encontradas en orden css. El parámetro rgb_small_frame es la imagen de ¼ de tamaño regenerada para optimizar rendimiento.

Función	Líneas de código	Utilidad en la aplicación
face_recognition.api.compare_faces	matches = face_recognition.compare_faces(self.known_face_encodings, face_encoding)	Una lista de valores Verdadero/Falso que indican qué known_face_encodings coinciden con la codificación de cara a comprobar.
face_recognition.api.face_distance	face_distances = face_recognition.face_distance(self.known_face_encodings, face_encoding)	Un ndarray numpy con la distancia para cada cara en el mismo orden que el array 'faces'.

Nota. La tabla muestra las líneas de código utilizadas con las correspondientes funciones de Face Recognition anteriormente mencionadas, además de su utilidad reflejada en la aplicación.

A modo de resumen hasta el momento se ha tratado todo lo necesario como librerías, componentes y pasos que se necesitan para dar vida a este módulo de reconocimiento facial diseñado el cual tiene el objetivo de realizar el reconocimiento facial de posibles Shoulder Surfers. Adicionalmente, se presenta el algoritmo el cual está dado de forma general el proceso que se realiza en el módulo diseñado de reconocimiento facial con el trabajo de OpenCV y Face Recognition.

Algoritmo: Detección y Reconocimiento de Rostros

Entradas: Fotos del rostro de los participantes.

Salidas: Detección y reconocimiento de personas.

Descripción del problema: Identificación de personas

Paso 1: Carga y extracción de data de imágenes.

Paso 2: Codificación de vectores de 128-d para cada imagen.

Paso 3: Apertura de entrada por video cámara.

Paso 4: Captura, decodificación y transformación del fotograma de video.

Paso 5: Optimizar fotograma por redimensión.

Paso 6: Conversión de formato al fotograma.

Paso 7: Encontrar todos los rostros.

Paso 8: Posicionamiento y proyección de caras.

Paso 9: Detección e identificación de personas.

Para terminar con esta sección, y con el fin de no extenderse en este tema y dar lo más claro, conciso e importante que es necesitado para la comprensión de esta parte fundamental de la aplicación, se dejó un tema por tratar, el cual se basa en cómo se pudo realizar la mitigación cuando es detectado el posible atacante o Shoulder Surfer, este tema será tratado en la siguiente sección de forma detallada, por lo que en el algoritmo anteriormente descrito no se ve reflejado dicha interacción con la mitigación.

Mitigación y funcionalidades adicionales

Una vez terminada la aplicación hasta el momento con todas sus utilidades e interfaces de navegación, y en especial su módulo de reconocimiento facial. Es importante desarrollar una forma en la que se pueda mitigar el peor de los escenarios en donde el atacante consigue exitosamente su cometido al robar toda la información personal posible de la víctima a su paso. La principal idea que se vino a la mente fue analizar paso a paso este escenario con el fin de detectar el punto crítico que abre las puertas a que se facilite la presencia de este ataque, con lo cual se determinó que el punto crítico reside en el dispositivo personal del cual se podría sustraer dicha información, en concreto uno de los componentes de dicho dispositivo, su “pantalla”, esto no solo fue previsto en el transcurso del desarrollo de este trabajo, ya que algunos estudios tales como (Brudy et al., 2014) y (Schneegass et al., 2022) donde para el primero se desarrolla un aplicativo el cual permita detectar y un ataque de Shoulder Surfing en escenarios públicos donde la víctima tiene que utilizar pantallas de un tamaño considerable

para realizar consultas de sus cuentas y datos personales, en este estudio este aplicativo que desarrollaron acortaba la visibilidad de la pantalla a una zona en concreto en la cual se trata de seguir la mirada del usuario con el objetivo de que terceras personas no pudiesen ver dicha información a completo. El segundo consiste en crear una carcasa adaptada a dispositivos móviles junto con un diseño personalizado para que cada usuario pueda usar su teléfono. Lo destacado es que esta carcasa incorpora una lente de ojo de pez en la cámara frontal, lo que amplía el campo de visión de la cámara de 60 a 90 grados, incluso hasta 180 grados con el propósito de aportar una mejor visión del agresor.

En busca de la manera con la que se pueda frenar el progreso del robo de información por Shoulder Surfing, algunas ideas se hicieron presente, ideas tales como bloquear la sesión de Windows, reducir al mínimo el brillo de la pantalla por un lapso de tiempo y apagar la pantalla, de las presentes opciones se decidió utilizar la de apagar la pantalla de modo que la pantalla del equipo se oscurezca de la misma forma que cuando entra el equipo en suspensión por falta de uso, esto es logrado gracias a la implementación de la librería llamada pywin32 la cual como se menciona en (Hammond, 2017/2023) ofrece extensiones o APIs de Windows desde Python. En la cual particularmente para este desarrollo se decidió utilizar la función win32api.SendMessage la cual permite enviar un mensaje al sistema operativo Windows. Por agregar, las otras funciones mencionadas a pesar de que no se las han utilizado en el producto final, residen en la aplicación en el caso de futuras implementaciones o mejoras que se puedan dar. A continuación, se puede ver el código de la función implementada y el funcionamiento de la aplicación final.

```
def turn_off_screen_windows():

    SC_MONITORPOWER = 0xF170
```

```
wg.SendMessage(wc.HWND_BROADCAST, wc.WM_SYSCOMMAND,
SC_MONITORPOWER, 2)
```

A pesar de que se deje libre la opción al usuario de realizar un ingreso por teclado o por el ratón con el fin de que pueda volver a encender la pantalla, se diseñó la aplicación con el objetivo de que mientras el atacante visualice la pantalla se vuelva a bloquear la misma una y otra vez hasta que el atacante deje de observar la pantalla y salga del campo de visión de la cámara del dispositivo, sin embargo, sí la pantalla del equipo se apaga de repente por sí misma no solventa el problema o indica al usuario del dispositivo que se encuentra bajo ataque, por lo cual se buscó una librería la cual permita realizar notificaciones y alertar al usuario para tomar las debidas precauciones del caso, la librería que se utilizo es Winotify que según (Syahputra, 2021/2023) se indica que es un módulo enteramente diseñado con Python para crear notificaciones para Windows, donde a su vez no requiere dependencias o requisitos lo que únicamente necesita es PowerShell instalado en la máquina, la cual todos los sistemas operativos Windows lo trae ya instalado por defecto, de modo que mediante el siguiente código se pueda obtener como resultado la Figura 25.

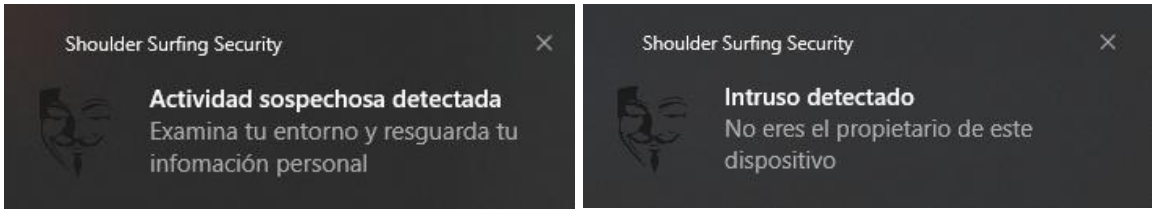
```
def _init_(self):
    self.toast = Notification(
        app_id="Shoulder Surfing Security",
        title="Actividad sospechosa detectada",
        msg="Examina tu entorno y resguarda tu infomación personal",
        duration="long",
        icon=os.path.join(self.RELATIVE_ASSETS_PATH,
"assets\images\winotify_icon.png"),
```

```
)

self.toast.set_audio(audio.Default, loop=False)
```

Figura 25

Notificación en Windows cuando se presenta un SSAs



Nota. La figura muestra la alerta de notificación de Windows cuando se presenta un ataque de SSAs y es detectado por la aplicación.

Una última pregunta que surge es ¿cómo se puede asegurar que un usuario pasa por un proceso de ataque? Para la cual primeramente se definirá 2 escenarios los cuales se ha definido como surfeo o navegación por los hombros (Shoulder Surfing) cuando se encuentre el propietario en uso del dispositivo y un atacante visualice su información con su mirada sobre el hombro de la víctima o detrás de ella, mientras que si el dispositivo se encuentra prendido y una persona ajena al mismo quiere ver la información personal de otro se denominará simplemente ataque. Con todo lo anteriormente visto en mente, la forma de asegurar estos casos es mediante algún identificador como se menciona en la sección Estado del Arte en el apartado de Factibilidad e impacto de la implementación del reconocimiento facial en la detección y mitigación de ataques de Shoulder Surfing, donde dicho identificador principalmente enlazara tanto al dispositivo como al propietario con un identificador único, de modo que estos sean los mismos al momento de compararlos, además se controló que se valide según el nombre del dispositivo el cual deberá de ser un conjunto de caracteres únicos que vinculen al dispositivo y la base de datos, mientras que para vincular el propietario con su rostro se asignó el nombre y apellido de la persona a la imagen sujeta al reconocimiento facial.

Capítulo IV: Aplicación y evaluación

Implementación de la Aplicación

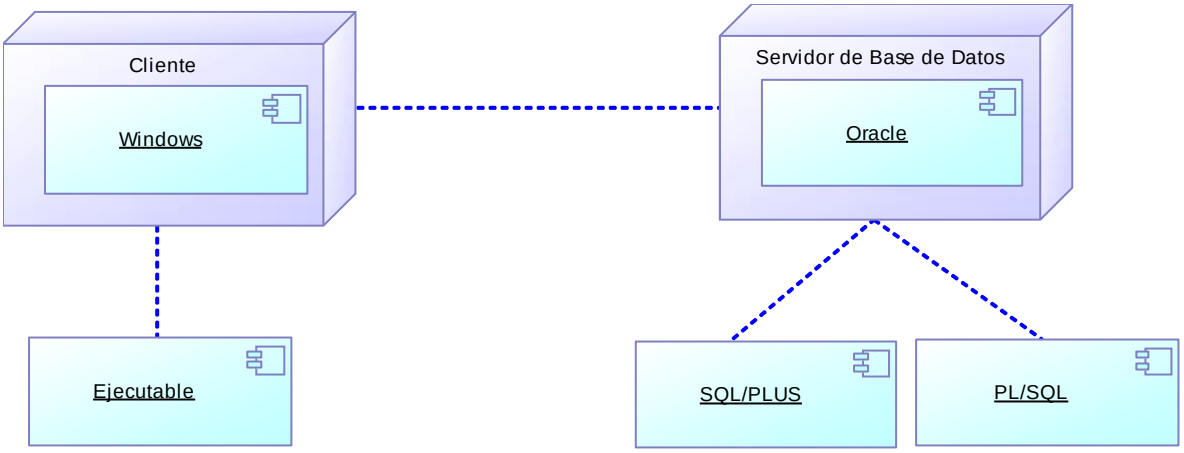
Instalación del Sistema

La aplicación tiene un proceso de instalación sencillo, ya que se proporciona el ejecutable que directamente el computador se abrirá y este tendrá una conexión a un servidor de base de datos como se muestra en la Figura 26**,** lo que sí es indispensable que el sistema operativo sea Windows 10 o Superior y que el equipo tenga una cámara previamente instalada.

Modelo de Despliegue

Figura 26

Diagrama de Despliegue



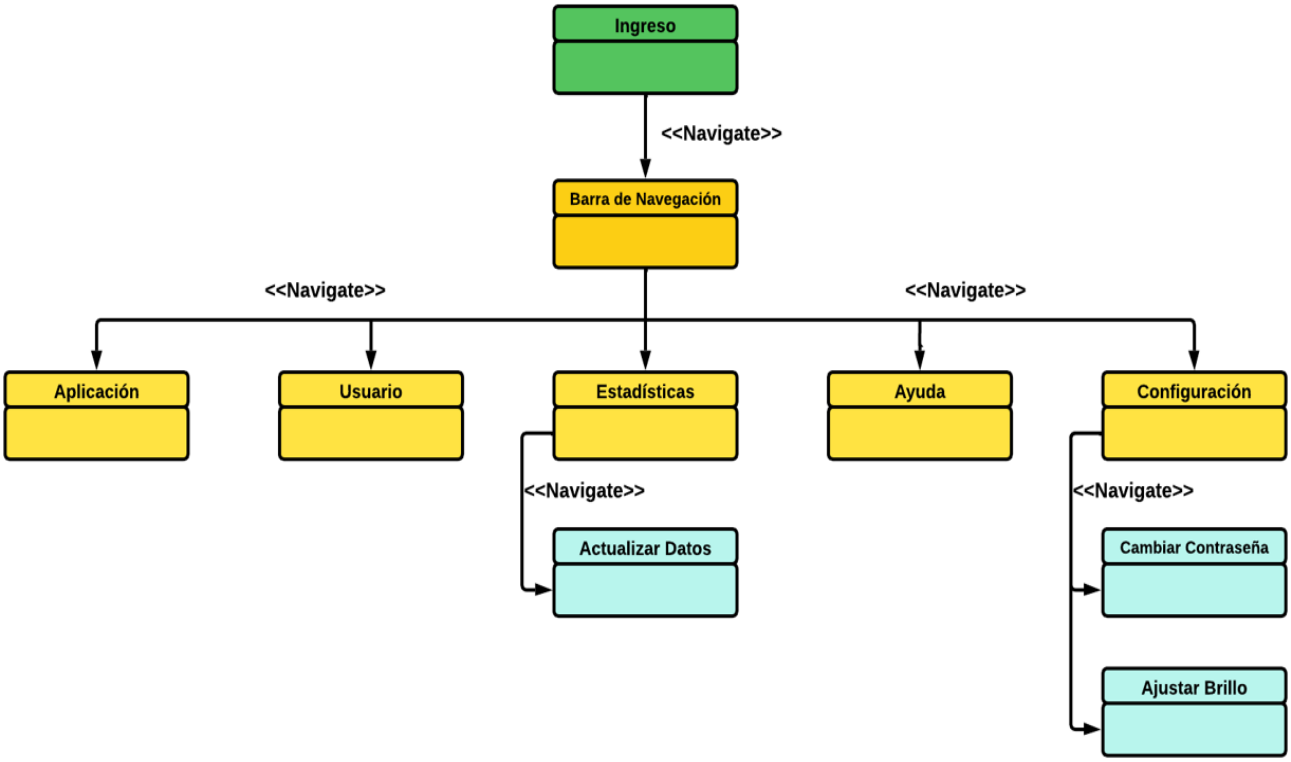
Nota. El grafico muestra la estructura del diagrama de despliegue

Modelo de Navegación

En la Figura 27 se puede observar la navegación que tiene la aplicación, representado principalmente por el correcto ingreso del usuario propietario para que pueda movilizarse y acceder a las diferentes vistas y herramientas.

Figura 27

Diagrama de Navegación



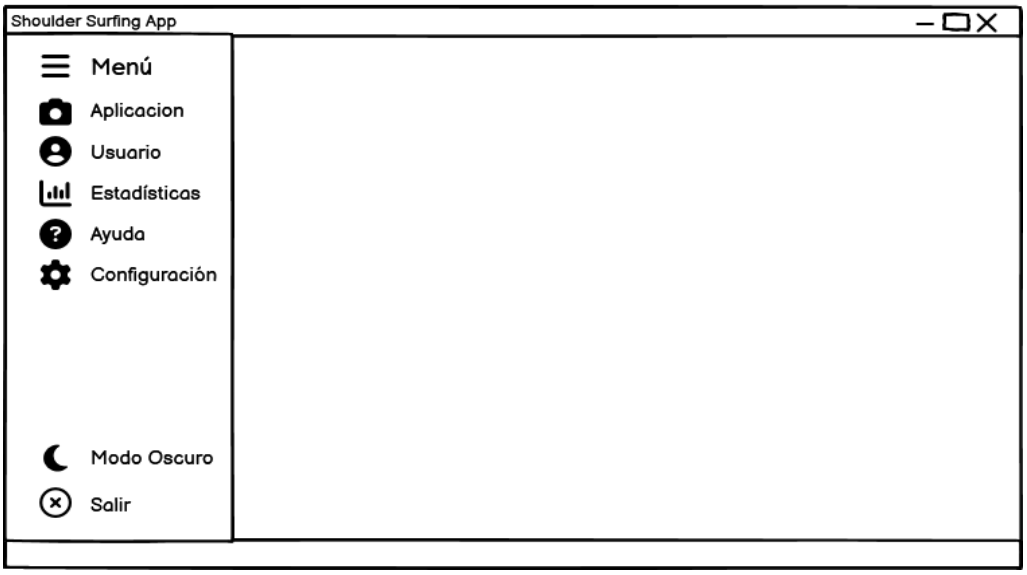
Nota. El grafico muestra la navegación que tiene el artefacto de software.

Diseño de Interfaz

El diseño de interfaz ofrece una perspectiva a futuro de cómo se vería la aplicación como tal, y es muy beneficiosa ya que nos brinda un lineamiento para poder codificar y preparar la parte gráfica del usuario. En la Figura 28, Figura 29, Figura 30, Figura 31, Figura 32, Figura 33, Figura 34 y Figura 35 se puede observar los bocetos de los diferentes escenarios que se tiene en la aplicación, la pantalla principal, el inicio de sesión, el error del inicio de sesión, al igual que las diferentes vistas que son la de aplicación, usuario, estadísticas, ayuda y configuraciones.

Figura 28

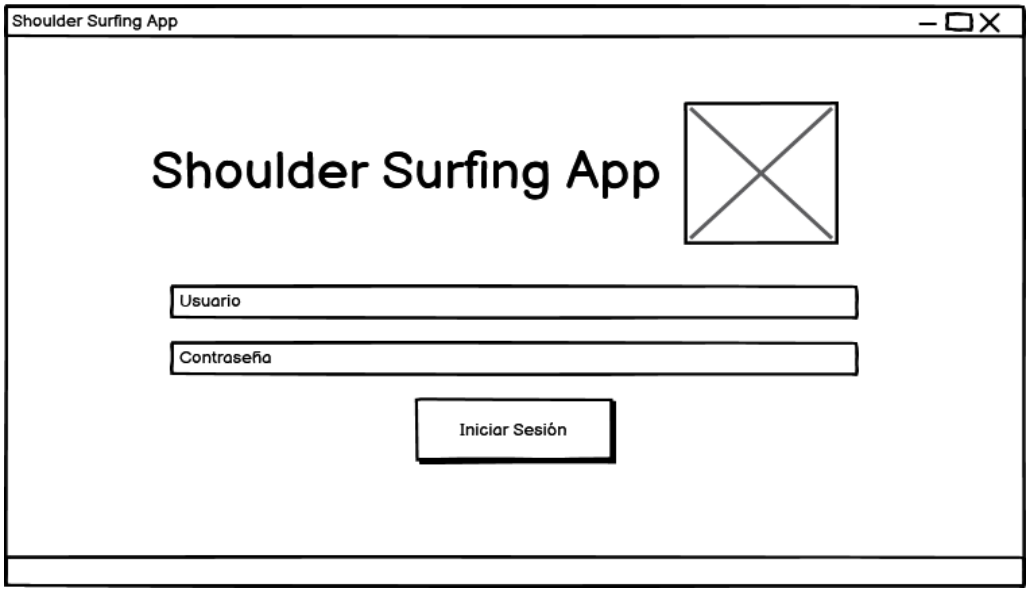
Boceto Pantalla Principal



Nota. El grafico muestra el boceto de la pantalla principal

Figura 29

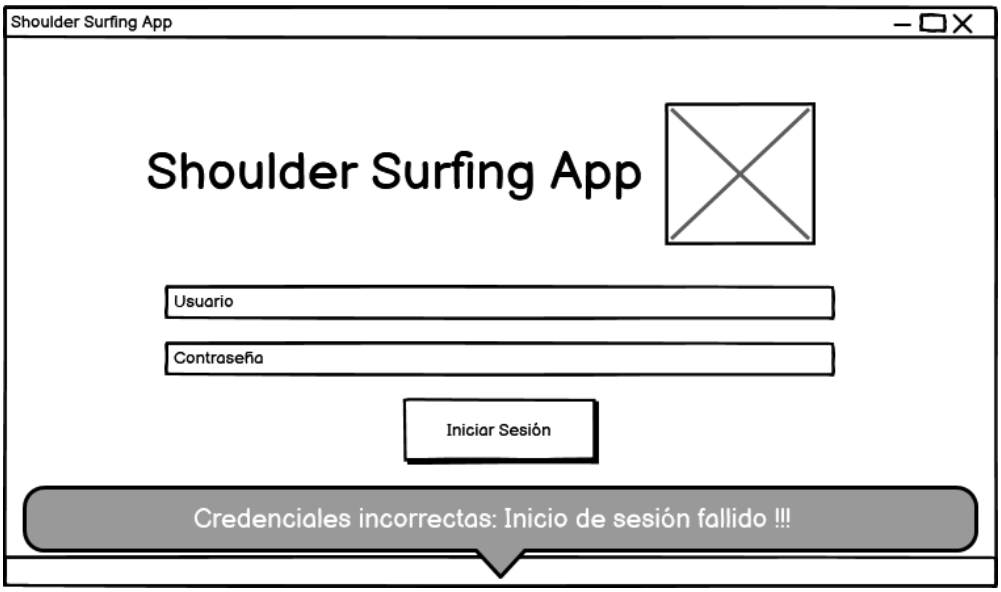
Boceto Iniciar Sesión



Nota. El grafico muestra el boceto del inicio de sesión

Figura 30

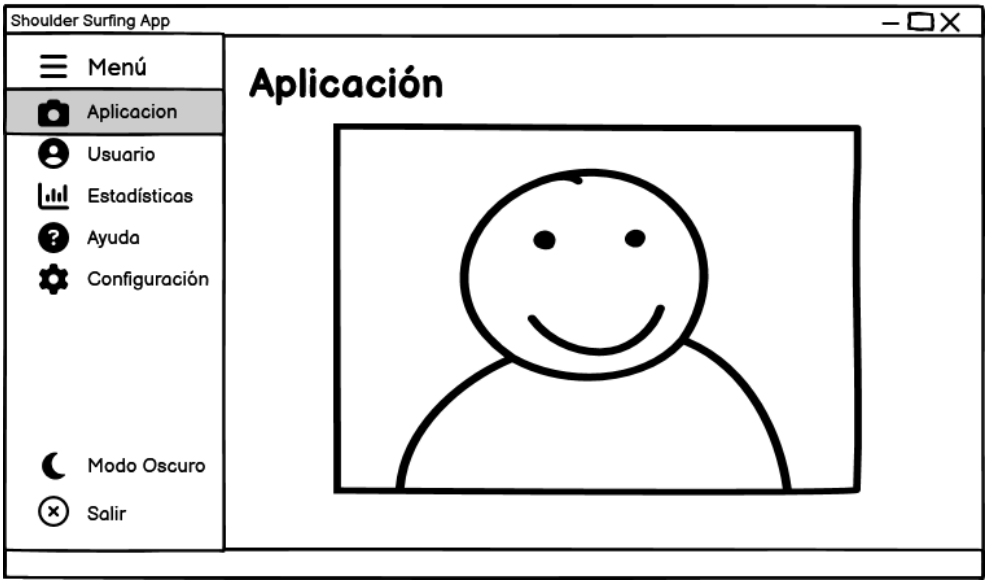
Boceto Iniciar Sesión Fallido



Nota. El grafico muestra el boceto del inicio de sesión fallido

Figura 31

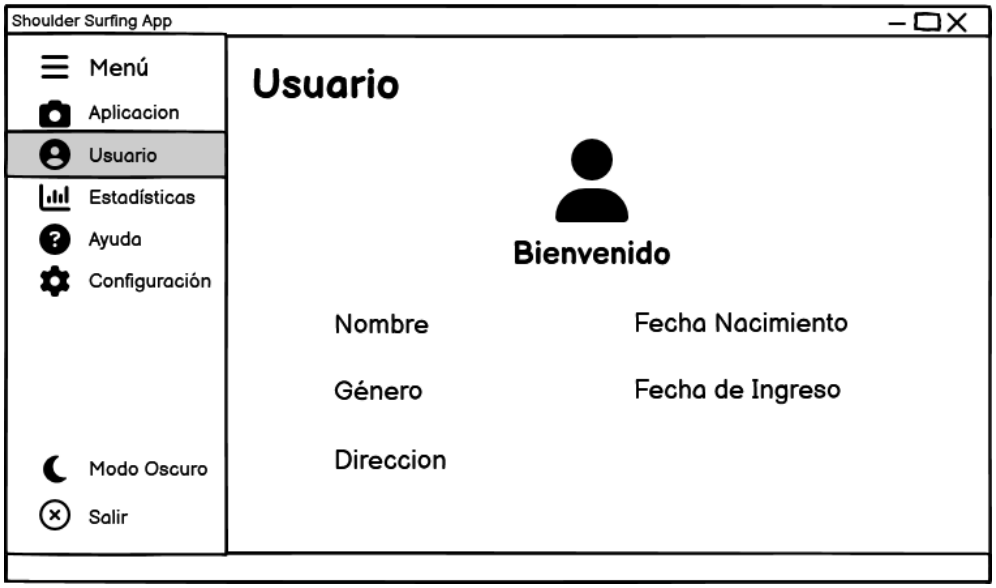
Boceto Vista de Aplicación



Nota. El grafico muestra el boceto de la vista de aplicación.

Figura 32

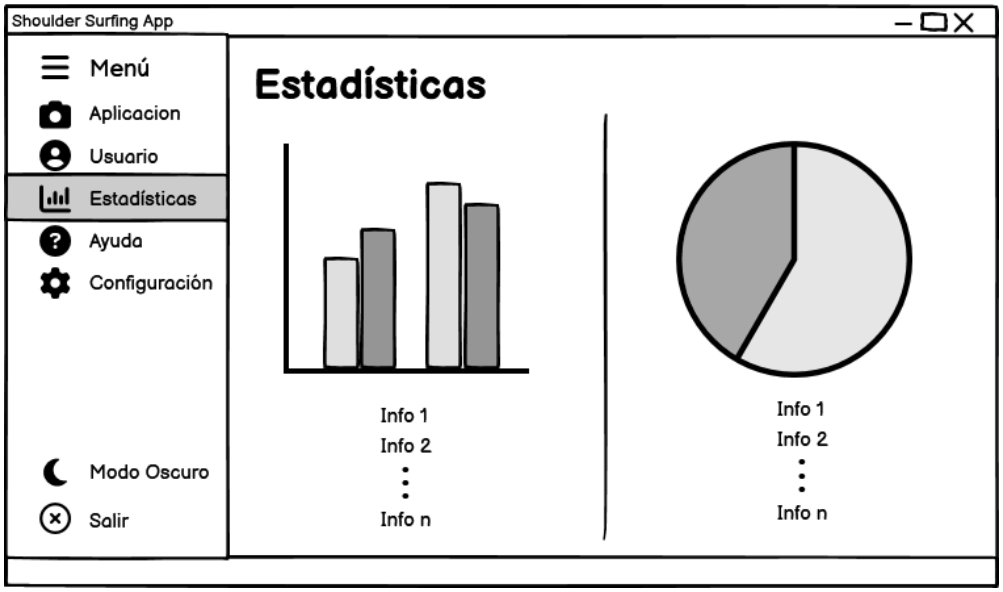
Boceto Vista de Usuario



Nota. El grafico muestra el boceto de la vista de usuario

Figura 33

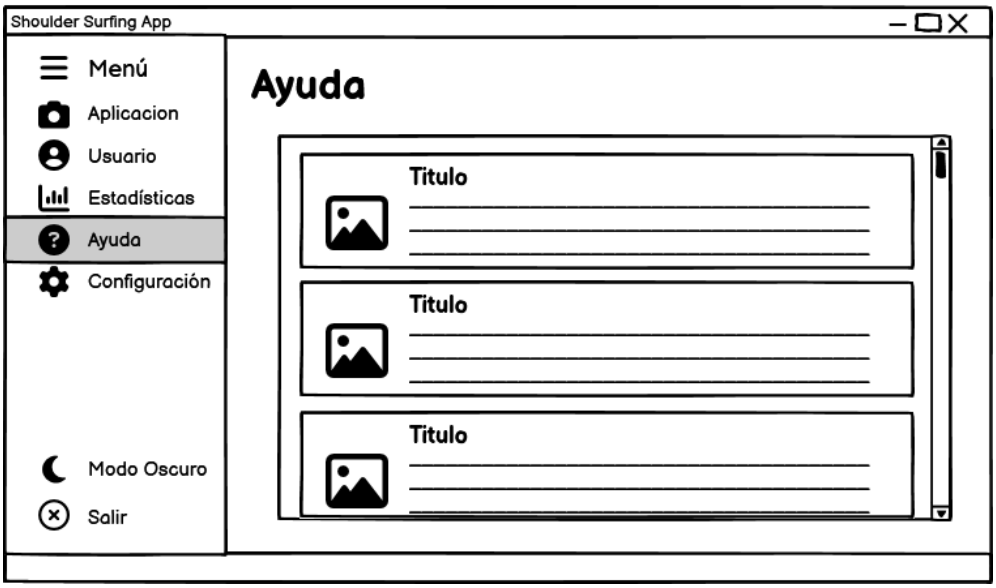
Boceto Vista de Estadísticas



Nota. El grafico muestra el boceto de la vista de estadísticas

Figura 34

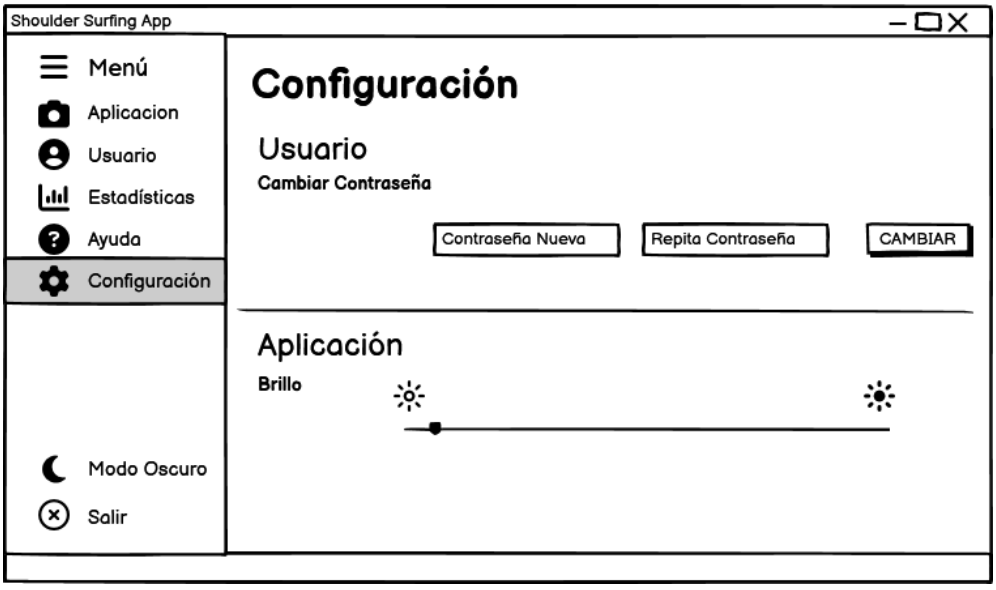
Boceto Vista de Ayuda



Nota. El grafico muestra el boceto de la vista de ayuda

Figura 35

Boceto Vista de Configuración



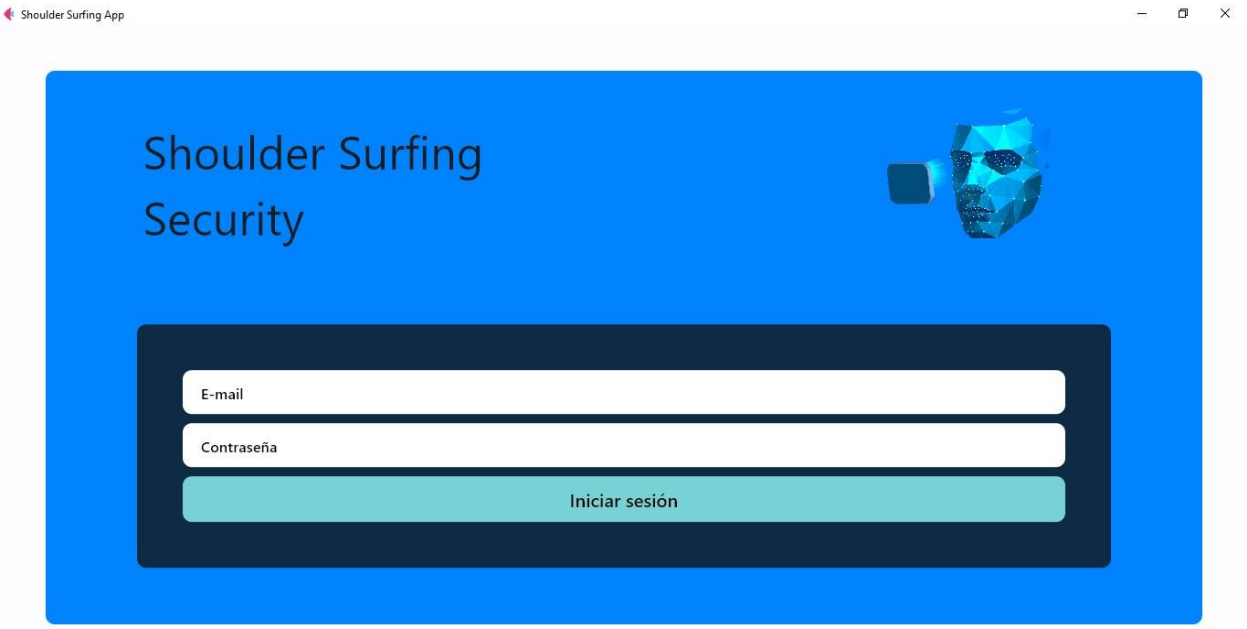
Nota. El grafico muestra el boceto de la vista de configuración

Puesta en marcha de la Aplicación

Al ejecutar la aplicación se puede observar que la funcionalidad y la interfaz que se proyectó cumplió con las expectativas, de manera alternada se presenta la aplicación en su ambiente de tema oscuro y tema normal, en la Figura 36, Figura 37, Figura 38, Figura 39, Figura 40, Figura 41 se puede observar la culminación del software y en la Figura 42 se puede observar los resultados estadísticos que serán detallados en la sección oportuna.

Figura 36

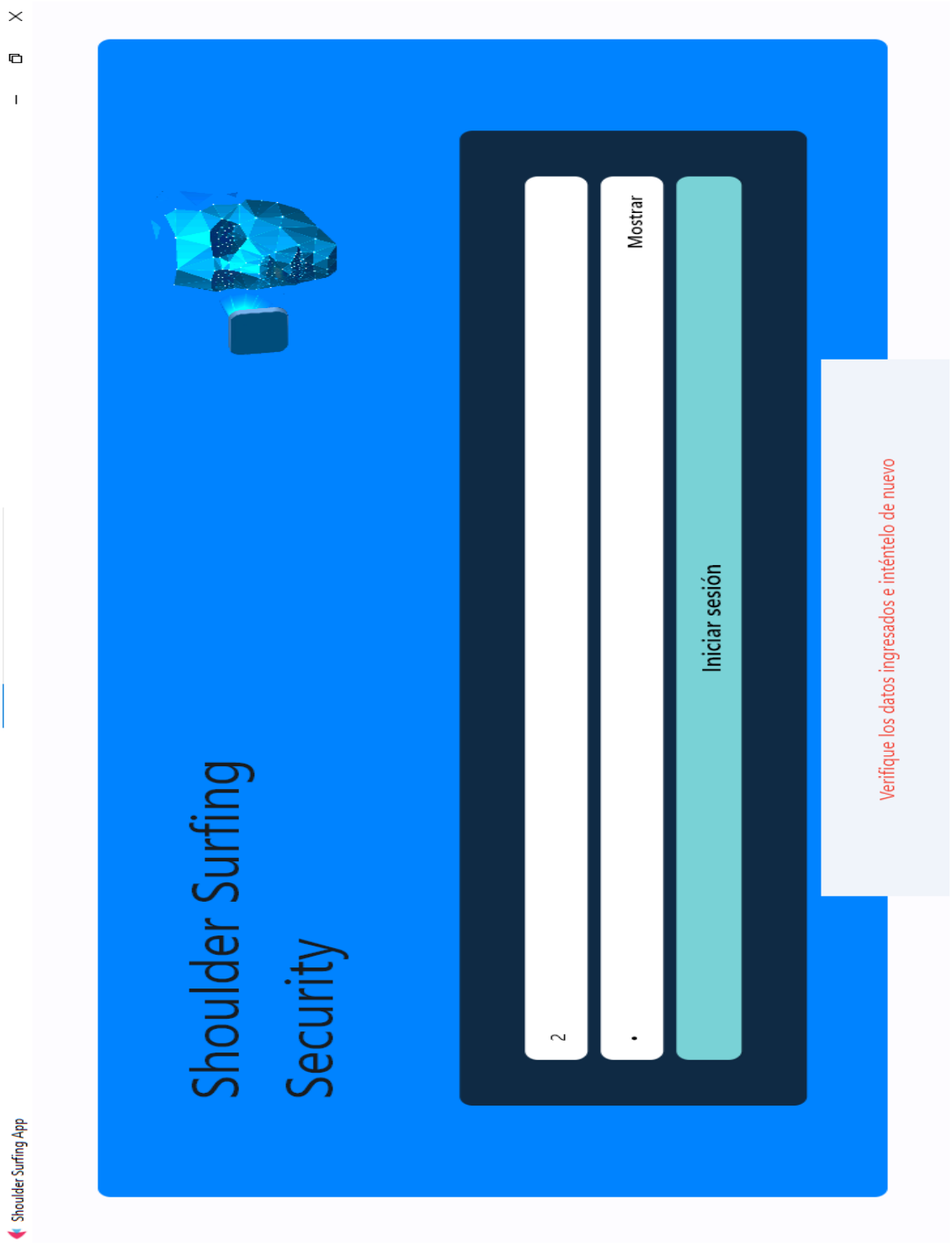
Aplicación: Iniciar Sesión



Nota. Vista final del inicio de sesión

Figura 37

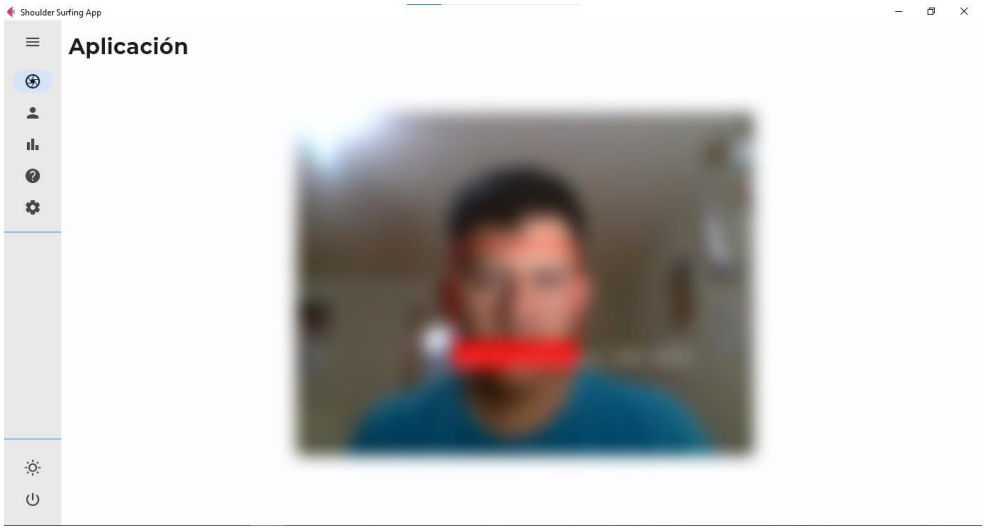
Aplicación: Iniciar Sesión Fallido



Nota. Vista final del inicio de sesión fallido

Figura 38

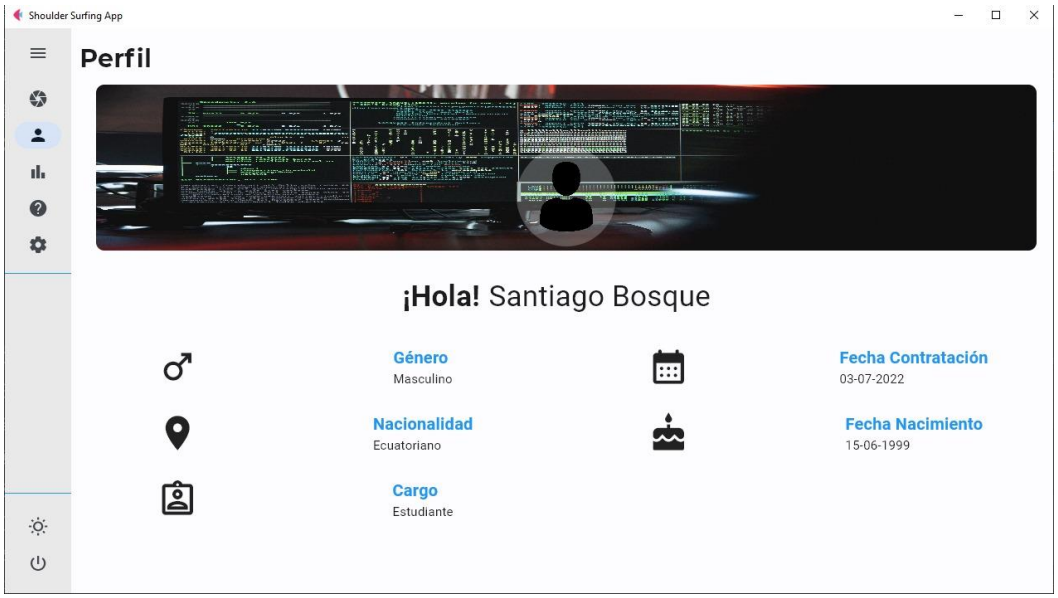
Aplicación: Detección Facial



Nota. Vista final de la detección facial.

Figura 39

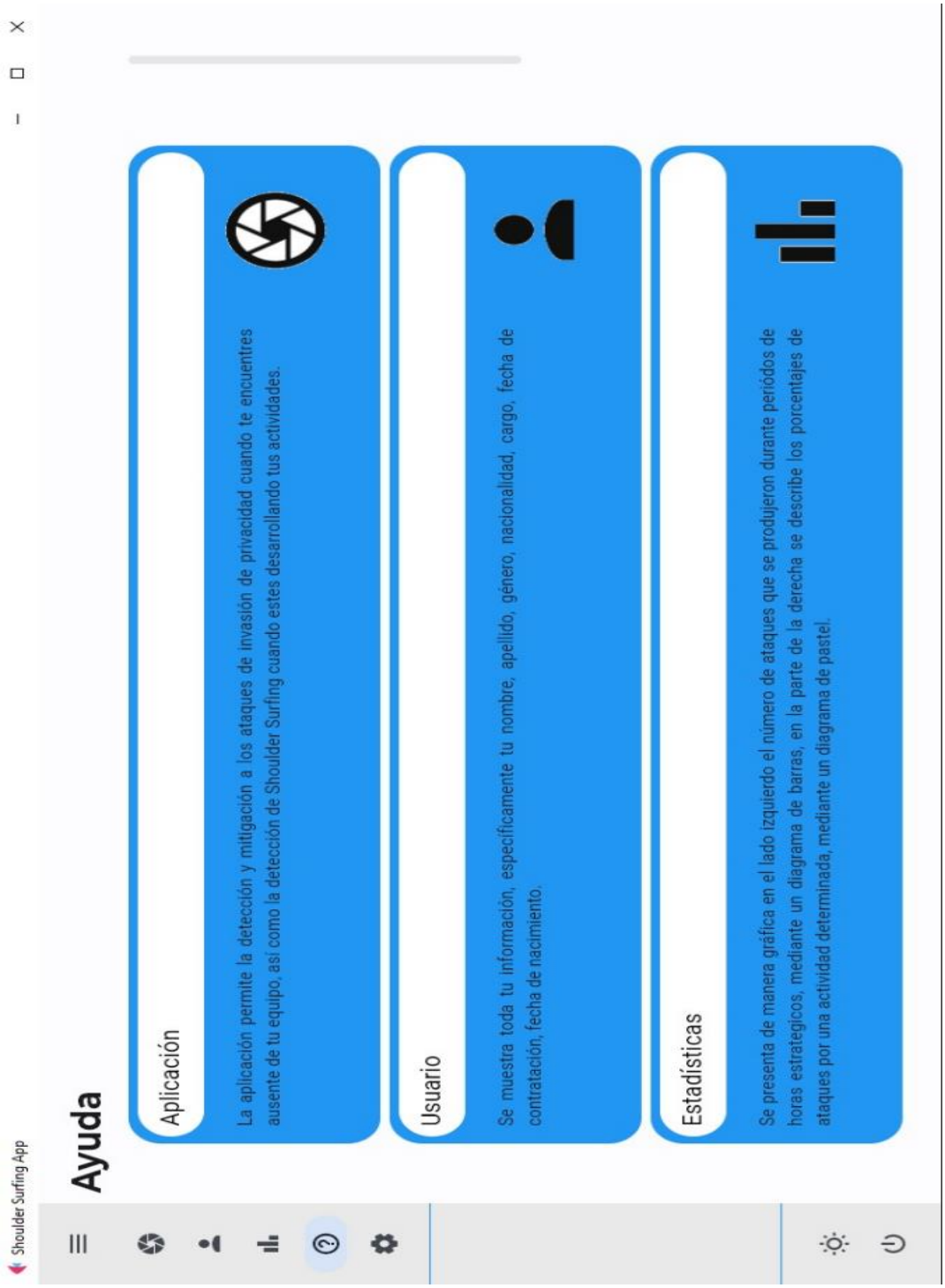
Aplicación: Perfil



Nota. Vista final del perfil de usuario

Figura 40

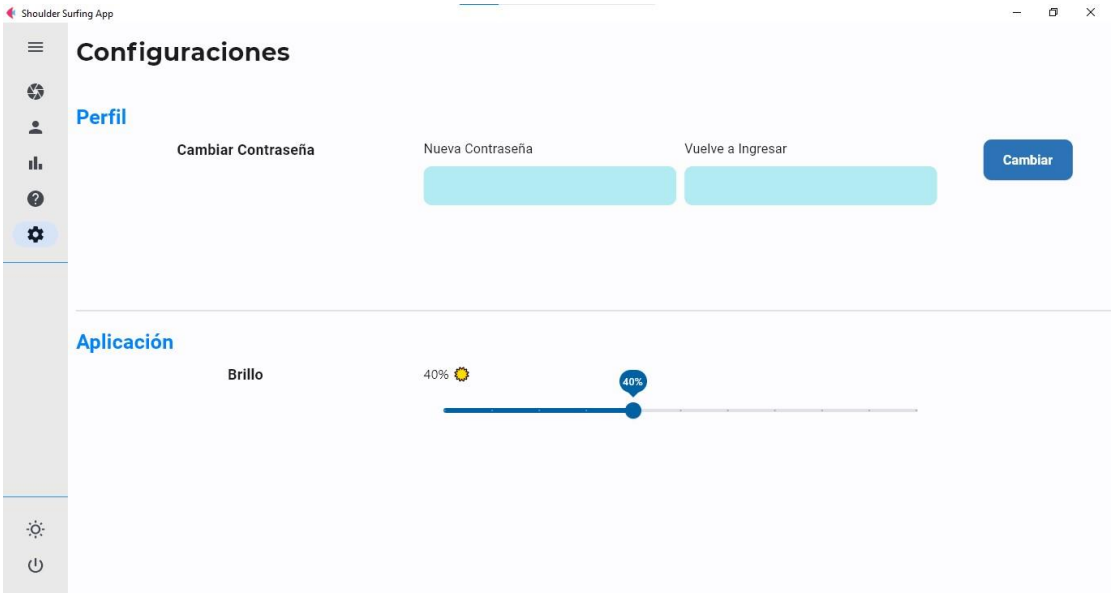
Aplicación: Ayuda



Nota. Vista final del apartado de ayuda.

Figura 41

Aplicación: Configuraciones



Nota. Vista final de las configuraciones

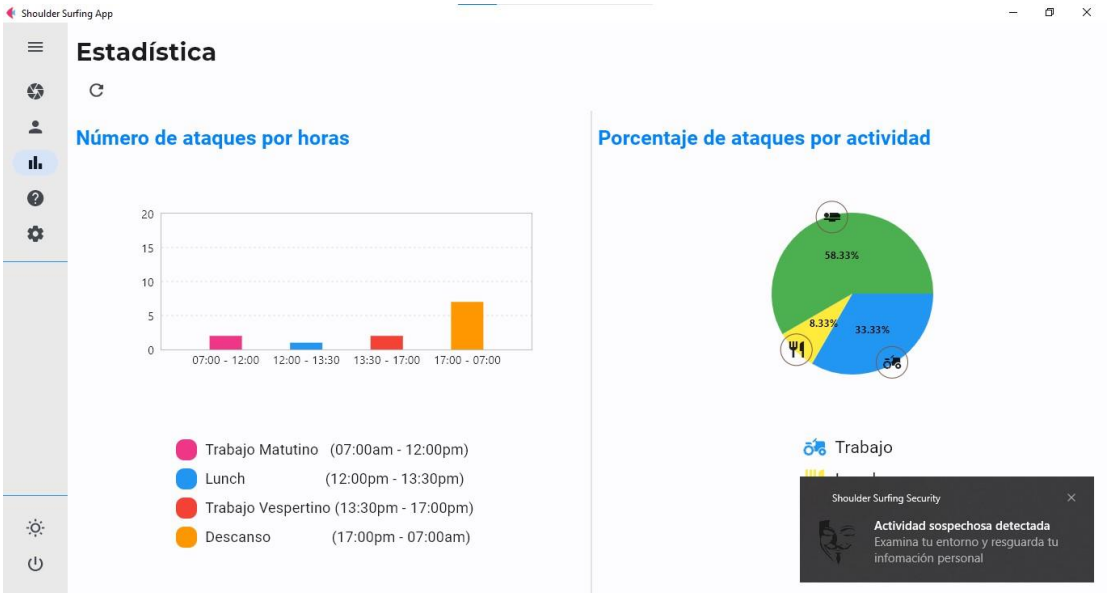
Generación de resultados mediante gráficos estadísticos

El poder obtener acceso a la información luego de un proceso previo puede ser confuso por el grado de detalles que se dé a los datos, talvez el visualizar extensos registros de tablas y datos innecesarios hacen que el interés sobre los resultados no tome gran relevancia, es por eso que se ha optado por utilizar gráficos estadísticos que puedan expresar de manera clara la información de resultados, esto con el fin de que la mayoría de usuarios puedan tener una interfaz amigable y sobre todo que puedan interpretar y entender los datos obtenidos

A continuación, se presenta en la Figura 42, como es la vista de los resultados estadísticos, y a continuación en la Figura 43 como se puede actualizar los datos, luego de haber sufrido un ataque.

Figura 42

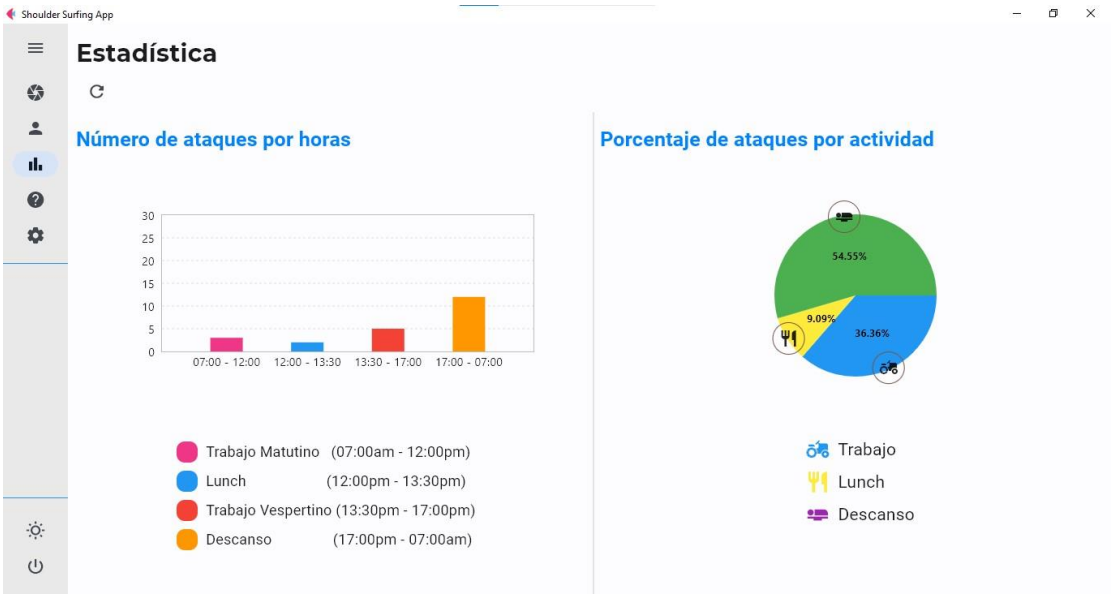
Imagen que muestra resultados estadísticos



Nota. Vista que muestra gráficos estadísticos y la notificación del ataque

Figura 43

Imagen que muestra resultados estadísticos luego de un ataque



Nota. Vista final de estadísticas luego de una actualización después del ataque

En la Figura 43 como se puede observar más a detalle la presentación estadística, en la parte izquierda se muestra un diagrama de barras que visualiza el número de ataques que se han realizado por horas, que se han definido de la siguiente manera, trabajo matutino de 7:00am a 12:00pm, lunch de 12:00pm a 13:30pm, trabajo vespertino de 13:30pm a 17:00pm, y por último el descanso de 17:00pm a 7:00am. Estos rangos horarios se han especificado por el estándar del ser humano en el que realiza sus actividades, claro está que no es algo totalmente cierto ya que dependería de la empresa y las actividades diarias que realiza. En la figura de la derecha tenemos un diagrama de pastel, que presenta el porcentaje de ataques por una actividad determinada, estos resultados se ayudan de la segmentación horario del anterior, lo que permite unificar el horario de trabajo matutino y vespertino, para denominarlo trabajo, por lo demás siguen la misma parametrización de lunch y descanso.

Para poder poner en ejecución estos lineamientos, se entregó una muestra de la aplicación a 41 personas que de manera voluntaria accedieron a colaborar con el estudio, en un periodo de 2 días en que la aplicación estaba en ejecución se pudo obtener los datos por hora que se muestra en la Tabla 11, y así mismo los porcentajes según la actividad que realiza un empleado común en su trabajo, que se muestra en la Tabla 12.

Tabla 11

Recolección y tabulación de datos de los participantes evaluados por hora

# Participante	7:00 – 12:00	12:00 – 13:30	13:30 - 17:00	17:00 – 7:00
1	3	5	6	10
2	2	2	7	9
3	4	4	4	6
4	5	4	5	12
5	1	3	4	7
6	2	7	4	10
7	3	5	7	9
8	4	4	3	9

# Participante	7:00 – 12:00	12:00 – 13:30	13:30 - 17:00	17:00 – 7:00
9	6	2	2	8
10	7	2	3	13
11	2	5	7	7
12	2	6	3	5
13	1	2	2	9
14	0	5	6	10
15	3	8	7	11
16	2	5	4	10
17	4	5	3	9
18	5	4	4	5
19	8	4	4	9
20	2	6	6	8
21	3	2	5	8
22	2	5	4	10
23	2	5	7	8
24	3	4	7	8
25	4	2	5	9
26	4	3	4	10
27	2	6	4	11
28	6	6	4	12
29	2	4	5	10
30	0	5	6	8
31	1	2	6	8
32	2	1	2	7
33	3	5	8	6
34	4	6	4	6
35	5	4	5	10
36	5	5	7	9
37	1	4	4	9
38	6	5	3	8
39	1	2	4	8
40	1	3	6	4
41	2	2	6	5
42	4	6	7	7
PROMEDIO	**3.07**	**4.16**	**4.85**	**8.5**

Nota. Tabla que muestra los datos obtenidos por segmentación de horas

Los resultados obtenidos en la tabulación de la Tabla 11 muestra a simple vista que el promedio más alto en que suceda un ataque es cuando no trabaja con aproximadamente 9 ataques cada 2 días, mientras que le siguen el trabajo en el horario de la tarde con aproximadamente 5 ataques cada 2 días, luego le sigue la hora del almuerzo con aproximadamente 4 ataques cada 2 días y por último el trabajo de la mañana con aproximadamente 3 ataques cada 2 días, esto puede ser determinado por varios factores como la presencia o no del propietario, la carga horaria que posea o por factores externos como recursos del computados o elementos ambientales. Cabe recalcar que esto es una simulación de cómo funciona la aplicación, son datos que se acercan a la realidad, sin embargo, no en su totalidad.

Tabla 12

Recolección y tabulación de datos de los participantes evaluados por actividad

# Participante	Trabajo (%)	Almuerzo (%)	Descanso (%)
1	37,5	20,83	41,66
2	45	10	45
3	44,44	22,22	33,33
4	38,46	15,38	46,15
5	33,33	20	46,66
6	26,08	30,43	43,47
7	41,66	20,83	37,5
8	35	20	45
9	44,44	11,11	44,44
10	40	8	52
11	42,85	23,80	33,33
12	31,25	37,5	31,25
13	21,42	14,28	64,28
14	28,57	23,80	47,61
15	34,48	27,58	37,93

# Participante	Trabajo (%)	Almuerzo (%)	Descanso (%)
16	28,57	23,80	47,61
17	33,33	23,80	42,85
18	50	22,22	27,77
19	48	16	36
20	36,36	27,27	36,36
21	44,44	11,11	44,44
22	28,57	23,80	47,61
23	40,90	22,72	36,36
24	45,45	18,18	36,36
25	45	10	45
26	38,09	14,28	47,61
27	26,08	26,08	47,82
28	35,71	21,42	42,85
29	33,33	19,04	47,61
30	31,57	26,31	42,10
31	41,17	11,76	47,05
32	33,33	8,33	58,33
33	50	22,72	27,27
34	40	30	30
35	41,66	16,66	41,66
36	46,15	19,23	34,61
37	27,77	22,22	50
38	40,90	22,72	36,36
39	33,33	13,33	53,33
40	50	21,42	28,57
41	53,33	13,33	33,33
42	45,83	25	29,16
PROMEDIO	**38.42**	**19.96**	**41.61**

Nota. Tabla que muestra los datos obtenidos por segmentación de actividad

Los resultados obtenidos en la tabulación de la Tabla 12 muestra que la actividad en donde se sufre más ataques es durante el descanso con el 41.61% mientras que cuando se realiza el trabajo tiene un 38.42% y en el lunch o almuerzo un 19.96%, una vez más se ratifica que el ausentarse del computador sin dejar resguardado o cerrado sesión es el escenario donde hay mayor vulnerabilidad y donde se produce el ataque, mientras que durante el trabajo hay colegas que intentan observar nuestro contenido, con intenciones que se desconocen y no se pueden asegurar, y en la hora de la comida claro está, el computador se queda sin su usuario propietario por lo que se presenta ataques de observación sin consentimiento.

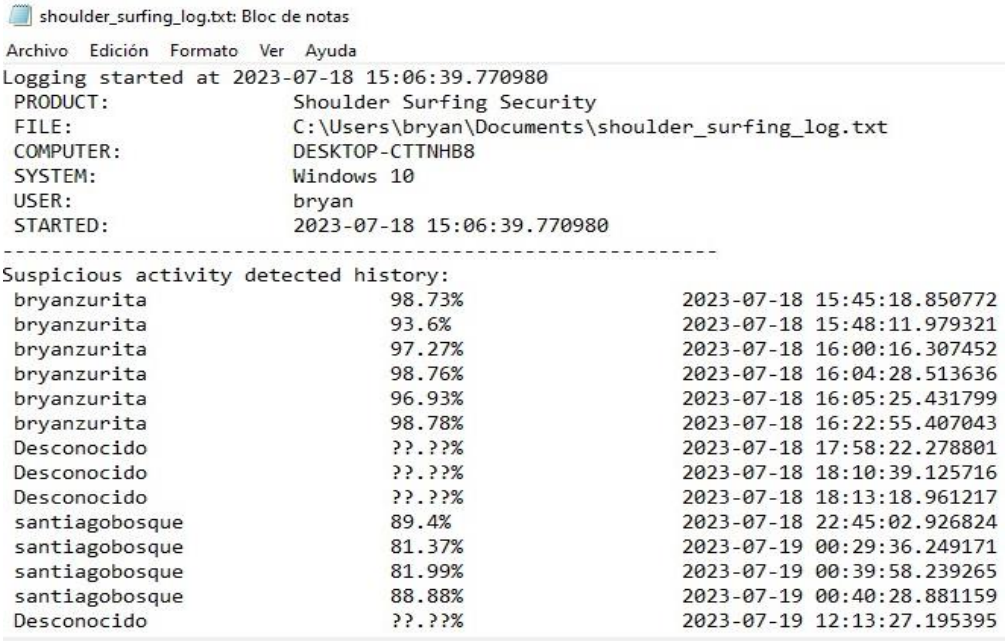
Generación de resultados mediante archivo log

La utilidad que puede representar un archivo de log en los sistemas aporta considerablemente a la seguridad de la información del propio sistema, esto también se puede ver en (Magomedov et al., 2021) donde se puede ver la implementación de este archivo para registrar eventos en una base de datos mientras se trabaja con servicios web a través de redes informáticas con registro de acciones de usuario como parte de la elaboración de un SIEM, acrónimo que corresponde a Security Information and Event Management al idioma inglés.

Con el fin de implementar el mismo concepto y promover la seguridad de la información, se decidió implementar un sistema de registros el cual tiene como objetivo generar un registro de todas las personas sospechosas a incurrir en un ataque de Shoulder Surfing. Estas quedarán registradas en un archivo de texto plano, el cual guardará el nombre y apellido del sospechoso en caso de ser una persona que se haya encontrado dentro del previo entrenamiento del reconocimiento facial, caso contrario se asignará la palabra Desconocido como se puede ver en la Figura 44, en la siguiente columna se presenta el porcentaje de precisión con la cual fue identificada dicha persona y en la columna restante se presenta la fecha y hora en la cual el posible Shoulder Surfer fue detectado.

Figura 44

Archivo de registro de actividad sospechosa de SSAs



shoulder_surfing_log.txt: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
Logging started at 2023-07-18 15:06:39.770980
 PRODUCT:              Shoulder Surfing Security
 FILE:                 C:\Users\bryan\Documents\shoulder_surfing_log.txt
 COMPUTER:             DESKTOP-CTTNHB8
 SYSTEM:               Windows 10
 USER:                 bryan
 STARTED:              2023-07-18 15:06:39.770980
------------------------------------------------------------
Suspicious activity detected history:
 bryanzurita                    98.73%                  2023-07-18 15:45:18.850772
 bryanzurita                    93.6%                   2023-07-18 15:48:11.979321
 bryanzurita                    97.27%                  2023-07-18 16:00:16.307452
 bryanzurita                    98.76%                  2023-07-18 16:04:28.513636
 bryanzurita                    96.93%                  2023-07-18 16:05:25.431799
 bryanzurita                    98.78%                  2023-07-18 16:22:55.407043
 Desconocido                    ??.??%                  2023-07-18 17:58:22.278801
 Desconocido                    ??.??%                  2023-07-18 18:10:39.125716
 Desconocido                    ??.??%                  2023-07-18 18:13:18.961217
 santiagobosque                 89.4%                   2023-07-18 22:45:02.926824
 santiagobosque                 81.37%                  2023-07-19 00:29:36.249171
 santiagobosque                 81.99%                  2023-07-19 00:39:58.239265
 santiagobosque                 88.88%                  2023-07-19 00:40:28.881159
 Desconocido                    ??.??%                  2023-07-19 12:13:27.195395
```

Nota. La figura muestra el archivo de registro log generado automáticamente mediante la aplicación, donde se almacenan los nombres de los atacantes con respecto del dispositivo vulnerado, porcentaje de precisión y fecha del SSAs.

El archivo mencionado se diseñó con la intención de que quedará almacenado en el dispositivo personal de la posible víctima en una ruta específica del mismo, el cual cada cierto tiempo todos los miembros de cada organización encargados de responder y salvaguardar la información mediante la implementación de los distintos protocolos, leyes y normativas de seguridad las cuales se encuentren implementadas en el Sistema de Gestión de Seguridad de la Información SGSI (*SGSI*, s. f.) de cada organización, estarán sujetos a llevar las correspondientes investigaciones del caso y manejar de forma responsable. Cabe mencionar que esta funcionalidad implementada difiere de la anterior especialmente en el tema relacionado con la protección de la identidad, ya que en un ambiente empresarial este tipo de información y datos, deberán de llevarse con toda la prudencia y cautela del caso. Para

terminar, la información relacionada a todos los requerimientos, procedimientos y conocimientos necesarios correspondientes al manejo de la aplicación serán indicados a las organizaciones para su correcto uso, por ejemplo, en este caso con la información referente a la localización de este archivo.

Encuesta de experiencia de usuario UX Nielsen y escala de Likert

El objetivo de esta evaluación es recopilar los datos relevantes de la aplicación, los cuales se califican según la experiencia del usuario o participante. Para la elaboración de esta encuesta se tomó en cuenta las 10 heurísticas de (Nielsen, 2005) propuestas para el análisis de los 10 principios generales para diseñar interfaces de usuario, hay que tomar en consideración que estas reglas son más empíricas que directrices específicas de usabilidad. Estas reglas, se detallan a continuación:

1. Visibilidad del estado del sistema: El sistema debe mantener informados a los usuarios sobre lo que ocurre en todo momento.
2. Correspondencia entre el sistema y el mundo real: El sistema debe hablar un idioma fácil y sencillo con palabras, frases y conceptos familiares para el usuario de modo que muestre la información en un orden natural y lógico.
3. Control y libertad del usuario: Los usuarios deben tener una "salida de emergencia" a una acción realizada, sin tener que pasar por un diálogo prolongado.
4. Consistencia y normas: El sistema deberá seguir convenciones de la plataforma para aclarar el significado de situaciones o acciones.
5. Prevención de errores: Se debe eliminar las condiciones propensas a errores, además se debe permitir a los usuarios tener una confirmación antes de que lleven a cabo la acción.
6. Reconocimiento en lugar de recuerdo: Minimizar la carga de memoria del usuario mediante instrucciones de uso del sistema visibles o fácilmente recuperables.

7. Flexibilidad y eficacia de uso: El sistema debe adaptarse tanto a usuarios inexpertos como experimentados.
8. Diseño estético y minimalista: Los diálogos y componentes deben tener información relevante, clara y concisa a modo que disminuya su visibilidad relativa.
9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores: El sistema debe mostrar errores con un lenguaje sencillo y preciso, además de sugerir una solución.
10. Ayuda y documentación: Aunque es propicio que el sistema pueda utilizarse sin documentación, puede ser necesario proporcionar ayuda y documentación. Dicha información debe ser fácil de buscar, centrarse en la tarea del usuario, enumerar los pasos concretos a realizar y no ser demasiado extensa.

La encuesta se encuentra dividida en cuatro subsecciones distribuidas de la siguiente manera: legibilidad (3 ítems), navegabilidad (3 ítems), usabilidad (2 ítems) y selección de contenidos (2 ítems); se buscó medir aspectos como la eficiencia del aplicativo y la utilidad del reconocimiento facial. Esta encuesta se enfocó en personas que tengan conocimiento de uso de las tecnologías de la información, principalmente el uso básico del computador donde para la elaboración de dicha encuesta dispondrán de aproximadamente 10 minutos; 1 minuto por pregunta para cada usuario con un total de 10 preguntas las cuales abarcaran todas o la mayoría de las temáticas anteriormente mencionadas.

A continuación, se muestra la tabulación de los resultados obtenidos por cada pregunta en la Tabla 13, donde cada pregunta se describe de la siguiente forma: i) ¿El diseño de la aplicación es lo suficientemente atractiva como para desear utilizarse en el día a día? (Q1); ii) ¿La navegación y los componentes de la aplicación se encuentran bien organizados y disponibles? (Q2); iii) ¿El reconocimiento facial se acopla al objetivo establecido en la aplicación? (Q3); iv) ¿Considera apropiada la implementación de estadísticas y registros en

archivos log? (Q4); v) El lenguaje utilizado, ¿Es conciso y concreto (no insinuante y ambiguo)? (Q5); vi) ¿Las fuentes y los colores se encuentran estandarizados? (Q6); vii) ¿La aplicación contiene texto e imágenes en un nivel equilibrado? (Q7); viii) ¿Si desea dirigirse a una sección específica de la aplicación lo haría sin mayor inconveniente? (Q8); ix) ¿Estaría dispuesto a recomendar esta aplicación a un conocido u organización? (Q9); x) ¿El acceso y la navegación a la aplicación resulta fácil? (Q10). Sin embargo, se debe señalar que los resultados fueron clasificados mediante el uso de la escala de Likert (Allen & Seaman, s. f.) la cual busca que los encuestados puedan encasillar la calidad de mayor a menor o de mejor a peor por medio del uso de cinco o siete niveles para varios tipos de datos entre los cuales se puede señalar a los datos nominales, datos ordinales, datos de intervalo y datos de relación, este último llegaría a ser el tipo de dato ideal para clasificar los resultados encontrados en este caso.

Tabla 13

Recolección y tabulación de datos obtenidos en la evaluación de UX

Opciones de respuesta	Q1	Q2	Q3	Q4	Q5
Totalmente De acuerdo	33,30%	72,10%	87,90%	68,20%	54,30%
De acuerdo	50,00%	22,00%	8,00%	11,17%	22,10%
Neutro	7,90%	3,60%	3,00%	3,12%	14,30%
Desacuerdo	6,70%	1,60%	1,10%	7,10%	3,60%
Totalmente Desacuerdo	2,10%	0,70%	0,00%	10,41%	5,70%

Opciones de respuesta	Q6	Q7	Q8	Q9	Q10
Totalmente De acuerdo	80,70%	72,17%	92,90%	35,00%	82,24%
De acuerdo	9,30%	19,30%	6,56%	42,90%	9,56%
Neutro	7,60%	4,30%	0,27%	14,30%	5,27%
Desacuerdo	1,10%	3,12%	0,17%	4,30%	1,77%
Totalmente Desacuerdo	1,30%	1,11%	0,10%	3,50%	1,16%

Nota. La tabla muestra los valores correspondientes a los resultados de las 41 encuestas que se realizaron a las personas participantes.

Análisis de tiempos de respuesta para la mitigación

El análisis con respecto al rendimiento de la mitigación que se diseñó para esta aplicación consta de la captura de 10 tiempos los cuales cada uno comienza a contabilizar desde que es reconocido el atacante por el reconocimiento facial hasta cuando la pantalla del dispositivo se apaga. Para diferir la obtención de resultados, se realizaron pruebas en 3 dispositivos distintos, los cuales se puede ver a detalle las características de cada uno en la Tabla 14.

Tabla 14

Características de los dispositivos utilizados para la evaluación de rendimiento

Dispositivo	Resolución de Webcam	Sistema operativo	CPU	Memoria	Disco
Acer Predator PH315-52	1280x720 HD	Windows 11 Home 64-bit	Intel Core i7-9750H - 2.60GHz	SODIMM 16GB - 2667 MHz	SSD NVMe - 250GB
LENOVO ideapad 330	640x480	Windows 10 Home 64-bit	Intel Core i3-8130U - 2.20GHz	SODIMM 12GB - 2133MHz	HDD - 1TB
HP 14-cm0xxx	640x360	Windows 10 Home Single Language 64-bit	AMD A9-9425 RADEON R5 - 3.1GHz	SODIMM 8GB - 1866MHZ	HDD - 1TB

Nota. La tabla presenta de manera resumida las características que presentan los dispositivos utilizados para la realización de la evaluación de rendimiento de la aplicación.

Con el fin de evitar posibles sesgos de datos se realizó este proceso 3 veces y se sacó el promedio de cada uno de los tiempos; es decir que se tomaron 10 tiempos para 3 corridas distintas de la aplicación, en donde al final de cada una de ellas se cerró la aplicación y se volvió a ejecutar, cabe mencionar que las fotografías con las cuales el reconocimiento facial ha sido entrenado para esta prueba fueron 2 con una resolución de 1280x720 HD cada una, una de ellas corresponde al propietario del dispositivo y la otra corresponde a la foto de un atacante en modo de prueba extraída desde la web a modo de prueba. Los tiempos recolectados para cada una de las corridas se presentan en la siguiente Tabla 15.

Tabla 15

Recolección y tabulación de datos para la evaluación de rendimiento

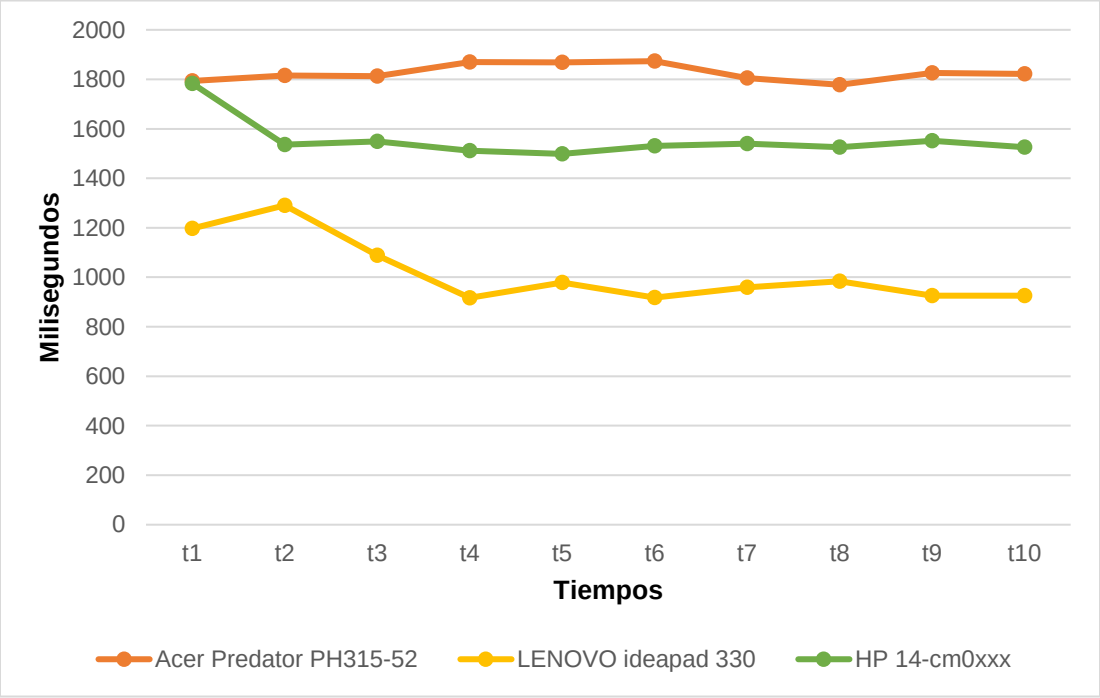
Dispositivo	#	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
Acer Predator PH315-52	1	1790	1811	1811	1943	1925	1915	1817	1686	1882	1870
	2	1800	1814	1817	1893	1905	1925	1828	1819	1817	1773
	3	1791	1822	1812	1775	1776	1780	1772	1829	1778	1824
	Avg	1793,66	1815,66	1813,33	1870,33	1868,66	1873,33	1805,66	1778	1825,66	1822,33
LENOVO ideapad 330	1	1371	1314	1012	927	1005	919	1003	951	917	908
	2	1298	1331	935	910	1020	920	958	916	936	932
	3	922	1227	1317	914	910	916	915	1083	925	936
	Avg	1197	1290,66	1088	917	978,33	918,33	958,66	983,33	926	925,33
HP 14-cm0xxx	1	1875	1594	1501	1519	1504	1517	1523	1507	1566	1519
	2	1764	1502	1511	1510	1505	1535	1545	1528	1548	1542
	3	1711	1512	1634	1506	1487	1539	1551	1544	1541	1518
	Avg	1783,3	1536	1548,66	1511,66	1498,66	1530,33	1539,66	1526,33	1551,66	1526,33

Nota. La tabla presenta de forma ordenada los datos generados a través de la evaluación de rendimiento realizada en los dispositivos anteriormente mencionados para cada uno de los tiempos y corridas de la aplicación.

El promedio dado para cada uno de los tiempos se presenta en la siguiente gráfica de la Figura 45.

Figura 45

Resultados de tiempos de respuesta en evaluación de rendimiento



Nota. El grafico presenta los resultados obtenidos tras la evaluación de rendimiento en los 3 dispositivos; los 10 tiempos son dados por los promedios de cada uno de los tiempos en base a las 3 corridas de la aplicación.

Tras el analizar la gráfica generada a partir de los resultados obtenidos en las 3 pruebas realizadas para cada dispositivo, se puede observar que el dispositivo el cual tiene menor tiempo de demora entre la identificación del atacante y la correspondiente mitigación es el dispositivo Lenovo ideapad 330, lo cual resulta un tanto curioso debido a que el esperado ganador en menor tiempo de demora sería el dispositivo Acer Predator PH315-52 debido a sus características, no obstante lo que sobresale es la resolución de su webcam de 1280x720 HD, mientras que la resolución de la webcam del dispositivo ganador es de 640x480.

Evidentemente este es el factor por el cual el tiempo entre el reconocimiento del atacante y la mitigación es más largo correspondiente debido al número de cálculos que se deben realizar para obtener las características de una imagen de esa resolución de píxeles, por otra parte, se puede mencionar que aquello correspondiente a fluidez de la aplicación y su uso el dispositivo Acer Predator es sin duda alguna el ganador.

Por último, algo importante que se pudo detectar durante y después de la evaluación son las características mínimas requeridas de hardware para que funcione correctamente la aplicación, estas características en realidad se centran principalmente en los componentes de video cámara, procesador y memoria, en el mismo orden de prioridad que se indican. La video cámara del dispositivo es un condicionante el cual determinará cuan altos deberán ser los niveles potencia, debido la resolución de la misma; es decir que a más resolución de calidad de la video cámara más pixeles tendrá que analizar el reconocimiento facial con el fin de comparar los rostros que entran por la imagen generada por la video cámara y la imagen con la cual es ingresada al reconocimiento facial para su entrenamiento. Con esto se quiere decir que si se tuviese un dispositivo el cual tenga una resolución aceptable de generación de imagen por videocámara como por ejemplo el dispositivo Lenovo ideapad 330 con una resolución de 640x480 sería más que suficiente tener un procesador que se maneje de 2.20GHz en adelante y una memoria de 8GB en adelante que por lo general todos los dispositivos actuales tienen este tipo de características e incluso superiores.

Análisis de tiempos de respuesta según el número de rostros

El análisis con respecto al rendimiento según el número de rostros que se diseñó para esta aplicación consta de la captura de 5 tiempos los cuales la aplicación prepara el repositorio de imágenes para su funcionamiento. Se puede observar el Figura 46 que los datos obtenidos en un determinado número de imágenes y en diferentes tiempos siguen una línea equilibrada, que aumenta según la cantidad de imágenes.

Tabla 16

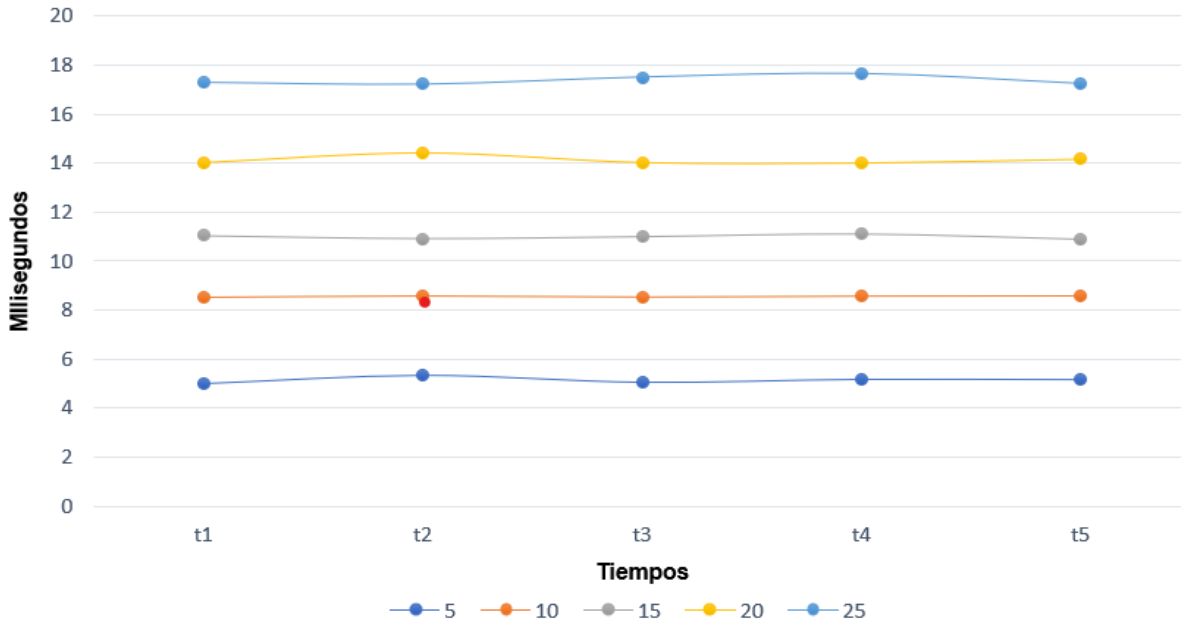
Recolección y tabulación de datos para la evaluación de cantidad imágenes

# Imágenes	t1	t2	t3	t4	t5	Promedio
5	5.01	5.32	5.06	5.17	5.16	5.14
10	8.51	8.60	8.53	8.59	8.61	8.57
15	11.05	10.92	11.01	11.13	10.89	11.00
20	14.02	14.43	14.02	14	14.16	14.13
25	17.29	17.22	17.5	17.64	17.25	17.38

Nota. La tabla muestras los datos de la cantidad de imágenes y el tiempo.

Figura 46

Resultados de tiempos de respuesta en evaluación de cantidad imágenes



Nota. El grafico muestra la representación de los tiempos según el número de imágenes

Tabla 17

Tabulación de datos de incremento de tiempo según la cantidad imágenes

# Imágenes	Promedio	Incremento según el Valor Anterior
5	5.14	
10	8.57	3.42
15	11.00	2.43
20	14.13	3.13
25	17.38	3.25
Promedio de Incremento	**≈**	**3.06**

Nota. El grafico muestra los promedios y el incremento según el valor anterior

En la Tabla 17 se puede observar que hay un promedio de crecimiento de 3.06 segundos por cada 5 imágenes que se aumenta, por lo que se interpretó que se podía hacer una fórmula matemática que pueda determinar el tiempo según de la cantidad de imágenes que se necesite. Esto da como resultado el siguiente análisis y su respectiva fórmula.

Tiempo ≈ Pendiente * Número de imágenes + (Intercepto – Promedio de Incremento)

Donde la pendiente sería 3.06 segundos / 5 imágenes = 0.612 segundos por imagen, el intercepto es el tiempo base que en este caso es 5.14 y el promedio de incremento es 3.06

Entonces, la fórmula sería:

Tiempo ≈ 0.612 * Número de imágenes + (5.14 – 3.06)

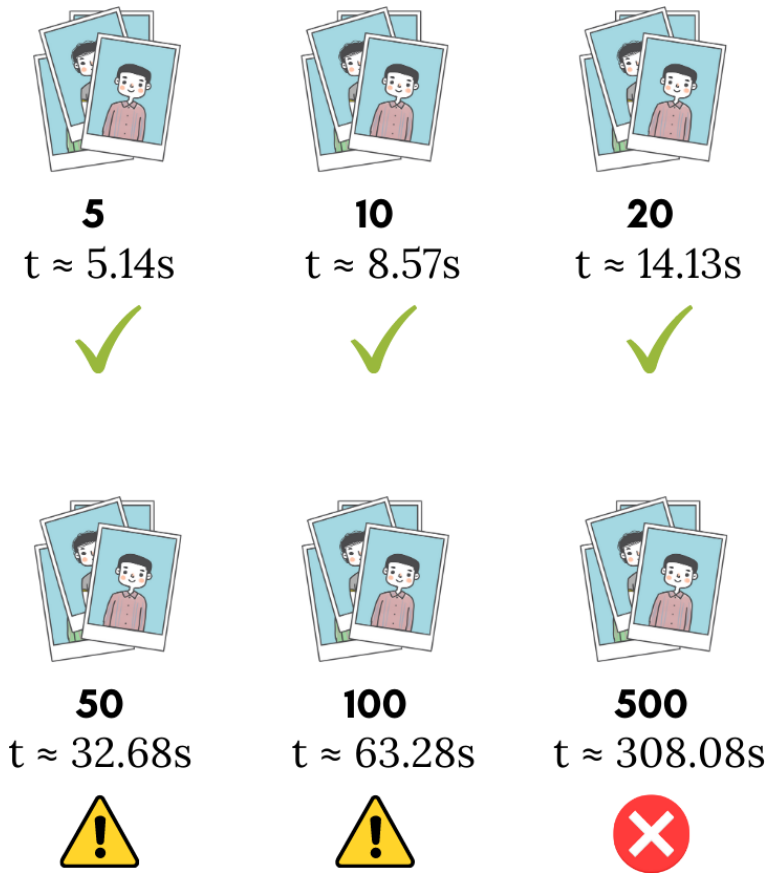
Ecuación 1

Fórmula para determinar el tiempo de respuesta de la aplicación según la cantidad de imágenes del dataset

$$\textbf{Tiempo} \approx \textbf{0.612} * \textbf{Número de imágenes} + \textbf{2.08}$$

Figura 47

Tiempos de respuesta según la cantidad de imágenes y su tipo de impacto



Nota. El grafico muestra los tiempos óptimos según la cantidad de imágenes

Se puede observar en la Figura 47 que mediante la fórmula obtenida se pudo obtener resultados aproximados de 50, 100 y 500 imágenes, la 50 imágenes dan como un promedio de 32.68 segundos que es un cantidad de tiempo considerable para esperar que una aplicación se

ejecute, mientras que 100 imágenes reflejan 63.28 segundos que ya sobrepasa el minuto y es preocupante, mientras que 500 imágenes reflejan 308.08 segundos que es algo alarmante ya que supera los 5 minutos te tiempo de espera, estos resultados ofrecen una guía del escenario de cantidad de imágenes donde puede funcionar la aplicación, en qué ambiente causaría problemas y una solución a la misma.

Capítulo V: Conclusiones y recomendaciones

Conclusiones

Con la implementación técnicas de Deep Learning se logró desarrollar un modelo de software el cual puede detectar y mitigar ataques de Ingeniería Social “Shoulder Surfing” permitiendo así aumentar los niveles de seguridad de la información.

Se encontró y analizo bases teóricas los cuales ayudan a consolidar los fundamentos correspondientes a la problemática y las posibles soluciones tecnológicas.

Con el apoyo de LFW Labeled Faces in the Wild se consiguió determinar la precisión y margen de error de los algoritmos de Deep Learning, además se diseñó la arquitectura de software incluyendo programas, algoritmos reutilizables, diseño de la base de datos, diseño de interfaces, usabilidad y navegabilidad.

Se realizaron pruebas del modelo de desarrollo de software y se realizó la documentación correspondiente según el progreso y las acciones.

Se evaluaron e interpretaron cada uno de los resultados obtenidos tras la validación e implantación del modelo de desarrollo de software.

Recomendaciones

Es recomendable que para obtener un correcto funcionamiento del aplicativo se lo implemente en un lugar adecuado, que tenga las condiciones de luz optimas, sistema de videovigilancia, estaciones de trabajo limpias sin obstáculos, y constante actualización de fotografías.

Se puede acoplar a otros ámbitos como el académico, siendo una herramienta que impida la copia en exámenes en línea o que una persona que no es la que deba rendir el examen pueda hacerlo.

Aplicar un constante versionamiento, en el cual se pueden implementar mejoras, corrección de errores, optimización de rendimiento y actualizaciones de seguridad.

Es indispensable realizar capacitaciones constantes en el ámbito tecnológico y socializar la variedad de artefactos y herramientas como la desarrollada en este estudio.

Referencias

14:00-17:00. (s. f.). *ISO/IEC 27001 Standard – Information Security Management Systems*. ISO. Recuperado 16 de agosto de 2023, de https://www.iso.org/standard/27001

267928931.pdf. (s. f.). Recuperado 16 de agosto de 2023, de https://core.ac.uk/download/pdf/267928931.pdf

Ali, M., Baloch, A., Waheed, A., Zareei, M., Manzoor, R., Sajid, H., & Alanazi, F. (2021). A Simple and Secure Reformation-Based Password Scheme. *IEEE Access*, *9*, 11655-11674. https://doi.org/10.1109/ACCESS.2020.3049052

Allen, I. E., & Seaman, C. A. (s. f.). *Likert Scales and Data Analyses*.

Baxter, A. L., BenZvi, S. Y., Bonivento, W., Brazier, A., Clark, M., Coleiro, A., Collom, D., Colomer-Molla, M., Cousins, B., Delgado Orellana, A., Dornic, D., Ekimtcov, V., ElSayed, S., Gallo Rosso, A., Godwin, P., Griswold, S., Habig, A., Hill, R., Horiuchi, S., … The SCiMMA and SNEWS Collaborations. (2022). Collaborative experience between scientific software projects using Agile Scrum development. *Software: Practice and Experience*, *52*(10), 2077-2096. https://doi.org/10.1002/spe.3120

Behera, S. K., Bhoi, S., Dogra, D. P., & Roy, P. P. (2018). Robustness Analysis of Motion Sensor Guided Air Authentication System. *IEEE Transactions on Consumer Electronics*, *64*(2), 171-179. https://doi.org/10.1109/TCE.2018.2843283

Benavides-Astudillo, E., Fuertes, W., Sanchez-Gordon, S., Nuñez-Agurto, D., & Rodríguez-Galán, G. (2023). A Phishing-Attack-Detection Model Using Natural Language Processing and Deep Learning. *Applied Sciences*, *13*(9), Article 9. https://doi.org/10.3390/app13095275

Benavides-Astudillo, E., Silva-Ordoñez, L., Rocohano-Rámos, R., Fuertes, W., Fernández-Peña, F., Sanchez-Gordon, S., & Bastidas-Chalan, R. (2022). Analysis of Vulnerabilities Associated with Social Engineering Attacks Based on User Behavior. En M. Botto-Tobar, S. Montes León, P. Torres-Carrión, M. Zambrano Vizuete, & B. Durakovic (Eds.),

Applied Technologies (pp. 351-364). Springer International Publishing. https://doi.org/10.1007/978-3-031-03884-6_26

Bošnjak, L., & Brumen, B. (2020). Shoulder surfing experiments: A systematic literature review. *Computers & Security*, *99*, 102023. https://doi.org/10.1016/j.cose.2020.102023

Brudy, F., Ledo, D., Greenberg, S., & Butz, A. (2014). Is Anyone Looking? Mitigating Shoulder Surfing on Public Displays through Awareness and Protection. *Proceedings of The International Symposium on Pervasive Displays*, 1-6. https://doi.org/10.1145/2611009.2611028

BrunelloN. (2021). *English: The image shows the basic structure of a deep neural network and the connections between layers. Part of the hidden layers is omitted, as both the number of them can vary depending on the requirements. The amount of neuronal units for each layer is variable, too, depending on the requirements, and the amount shown is only for example purposes.* Own work. https://commons.wikimedia.org/wiki/File:Example_of_a_deep_neural_network.png

Cárabe, L., & Cermeño, E. (2021). Stegano-Morphing: Concealing Attacks on Face Identification Algorithms. *IEEE Access*, *9*, 100851-100867. https://doi.org/10.1109/ACCESS.2021.3088786

Chan, P. P. K., Liu, W., Chen, D., Yeung, D. S., Zhang, F., Wang, X., & Hsu, C.-C. (2018). Face Liveness Detection Using a Flash Against 2D Spoofing Attack. *IEEE Transactions on Information Forensics and Security*, *13*(2), 521-534. https://doi.org/10.1109/TIFS.2017.2758748

Dimiccoli, M., Marín, J., & Thomaz, E. (2018). Mitigating Bystander Privacy Concerns in Egocentric Activity Recognition with Deep Learning and Intentional Image Degradation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *1*(4), 132:1-132:18. https://doi.org/10.1145/3161190

Dlib C++ Library. (s. f.). Recuperado 14 de agosto de 2023, de http://dlib.net/

Eiband, M., Khamis, M., von Zezschwitz, E., Hussmann, H., & Alt, F. (2017). Understanding Shoulder Surfing in the Wild: Stories from Users and Observers. *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 4254-4265. https://doi.org/10.1145/3025453.3025636

Emerson, S., Emerson, K., & Fedorczyk, J. (2021). Computer workstation ergonomics: Current evidence for evaluation, corrections, and recommendations for remote evaluation. *Journal of Hand Therapy*, *34*(2), 166-178. https://doi.org/10.1016/j.jht.2021.04.002

Farzand, H., Bhardwaj, K., Marky, K., & Khamis, M. (2021). The Interplay between Personal Relationships & Shoulder Surfing Mitigation. *Proceedings of Mensch und Computer 2021*, 338-343. https://doi.org/10.1145/3473856.3474006

Figueroa-Suárez, J. A., Rodríguez-Andrade, R. F., Bone-Obando, C. C., & Saltos-Gómez, J. A. (2018). La seguridad informática y la seguridad de la información. *Polo del Conocimiento*, *2*(12), Article 12. https://doi.org/10.23857/pc.v2i12.420

Flet. (2023). [Python]. Flet. https://github.com/flet-dev/flet (Obra original publicada en 2022)

Flutter—Crea hermosas aplicaciones nativas en tiempo récord. (s. f.). Recuperado 14 de agosto de 2023, de https://flutter-es.io/

Geitgey, A. (2023). *Face Recognition* [Python]. https://github.com/ageitgey/face_recognition (Obra original publicada en 2017)

Geitgey, A. (2020, septiembre 24). Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning. *Medium*. https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78

Hammond, M. (2023). *Pywin32* [C++]. https://github.com/mhammond/pywin32 (Obra original publicada en 2017)

How neural networks work—A simple introduction. (2011, marzo 5). Explain that Stuff. http://www.explainthatstuff.com/introduction-to-neural-networks.html

Hussain, S. A., & Balushi, A. S. A. A. (2020). A real time face emotion classification and recognition using deep learning model. *Journal of Physics: Conference Series*, *1432*(1), 012087. https://doi.org/10.1088/1742-6596/1432/1/012087

IBM Security Verify—Autenticación avanzada. (2022, septiembre 1). https://www.ibm.com/es-es/products/verify-identity/advanced-authentication

Ibrahim, T. M., Abdulhamid, S. M., Alarood, A. A., Chiroma, H., Al-garadi, M. A., Rana, N., Muhammad, A. N., Abubakar, A., Haruna, K., & Gabralla, L. A. (2019). Recent advances in mobile touch screen security authentication methods: A systematic literature review. *Computers & Security*, *85*, 1-24. https://doi.org/10.1016/j.cose.2019.04.008

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685-695. https://doi.org/10.1007/s12525-021-00475-2

JAVIER, A. B. (2008). *Seguridad de la información. Redes, informática y sistemas de información*. Ediciones Paraninfo, S.A.

Khan, M., Chakraborty, S., Astya, R., & Khepra, S. (2019). Face Detection and Recognition Using OpenCV. *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 116-119. https://doi.org/10.1109/ICCCIS48478.2019.8974493

King, D. (s. f.). *High Quality Face Recognition with Deep Metric Learning*. Recuperado 14 de agosto de 2023, de http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html

Learned-Miller, E., Huang, G. B., RoyChowdhury, A., Li, H., & Hua, G. (2016). Labeled Faces in the Wild: A Survey. En M. Kawulok, M. E. Celebi, & B. Smolka (Eds.), *Advances in Face Detection and Facial Image Analysis* (pp. 189-248). Springer International Publishing. https://doi.org/10.1007/978-3-319-25958-1_8

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444. https://doi.org/10.1038/nature14539

LEY_ORGÁNICA_DE_GESTIÓN_DE_LA_IDENTIDAD_Y_DATOS_CI_751. (s. f.).

Li, L., Mu, X., Li, S., & Peng, H. (2020). A Review of Face Recognition Technology. *IEEE Access*, *8*, 139110-139120. https://doi.org/10.1109/ACCESS.2020.3011028

Libro-del-curso.pdf. (s. f.). Recuperado 16 de agosto de 2023, de https://ocw.ehu.eus/pluginfile.php/40137/mod_resource/content/1/redes_neuro/contenidos/pdf/libro-del-curso.pdf

Los gestores de bases de datos (SGBD) más usados. (2019, abril 16). *Canal Informática y TICS*. https://www.inesem.es/revistadigital/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/

Magomedov, S., Ilin, D., & Nikulchev, E. (2021). Resource Analysis of the Log Files Storage Based on Simulation Models in a Virtual Environment. *Applied Sciences*, *11*(11), Article 11. https://doi.org/10.3390/app11114718

MaxWellBorn. (2019). *Italiano: Postura corretta al PC*. Own work. https://commons.wikimedia.org/wiki/File:Videoterminale.jpg#metadata

Miraftabzadeh, S. A., Rad, P., Choo, K.-K. R., & Jamshidi, M. (2018). A Privacy-Aware Architecture at the Edge for Autonomous Real-Time Identity Reidentification in Crowds. *IEEE Internet of Things Journal*, *5*(4), 2936-2946. https://doi.org/10.1109/JIOT.2017.2761801

Ongsulee, P. (2017). Artificial intelligence, machine learning and deep learning. *2017 15th International Conference on ICT and Knowledge Engineering (ICT&KE)*, 1-6. https://doi.org/10.1109/ICTKE.2017.8259629

OpenCV: OpenCV modules. (s. f.). Recuperado 15 de agosto de 2023, de https://docs.opencv.org/4.8.0/index.html

ORACLE Base de datos: ¿Por qué las empresas la prefieren? (s. f.). Recuperado 19 de agosto de 2023, de https://www.inventiva.net/blog/por-que-empresas-prefieren-base-de-datos-oracle

Oracle Database: Definición y funcionamiento. (2022, febrero 16). IONOS Digital Guide. https://www.ionos.es/digitalguide/hosting/cuestiones-tecnicas/oracle-database/

Physical Access Control—An overview | ScienceDirect Topics. (s. f.). Recuperado 16 de agosto de 2023, de https://www.sciencedirect.com/topics/computer-science/physical-access-control

Physical Security Control—An overview | ScienceDirect Topics. (s. f.). Recuperado 16 de agosto de 2023, de https://www.sciencedirect.com/topics/computer-science/physical-security-control

Qin, L., Peng, F., Long, M., Ramachandra, R., & Busch, C. (2021). Vulnerabilities of Unattended Face Verification Systems to Facial Components-based Presentation Attacks: An Empirical Study. *ACM Transactions on Privacy and Security*, *25*(1), 4:1-4:28. https://doi.org/10.1145/3491199

Rojas-Díaz, J. S., & Yepes-Londoño, J. J. (2022). Panorama de riesgos por el uso de la tecnología en América Latina. *Trilogía Ciencia Tecnología Sociedad*, *14*(26), e2020. https://doi.org/10.22430/21457778.2020

Romero Castro, M. I., Figueroa Morán, G. L., Vera Navarrete, D. S., Álava Cruzatty, J. E., Parrales Anzúles, G. R., Álava Mero, C. J., Murillo Quimiz, Á. L., & Castillo Merino, M. A. (2018). *Introducción a la seguridad informática y el análisis de vulnerabilidades* (1.ª ed.). Editorial Científica 3Ciencias. https://doi.org/10.17993/IngyTec.2018.46

Rosebrock, A. (2018, junio 18). Face recognition with OpenCV, Python, and deep learning. *PyImageSearch*. https://pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/

Ruiz, C. A., Basualdo, M. S., & Matich, D. J. (s. f.). *Redes Neuronales: Conceptos Básicos y Aplicaciones.*

Schneegass, S., Saad, A., Heger, R., Delgado Rodriguez, S., Poguntke, R., & Alt, F. (2022). An Investigation of Shoulder Surfing Attacks on Touch-Based Unlock Events. *Proceedings*

of the ACM on Human-Computer Interaction, *6*(MHCI), 207:1-207:14. https://doi.org/10.1145/3546742

SGSI. (s. f.). Recuperado 14 de agosto de 2023, de https://www.iso27000.es/sgsi.html

Shao, X.-F., Li, Y., Suseno, Y., Li, R. Y. M., Gouliamos, K., Yue, X.-G., & Luo, Y. (2021). How does facial recognition as an urban safety technology affect firm performance? The moderating role of the home country's government subsidies. *Safety Science*, *143*, 105434. https://doi.org/10.1016/j.ssci.2021.105434

Shreya, D. S. (2021). DIGITAL IMAGE PROCESSING AND RECOGNITION USING PYTHON. *International Journal of Engineering Applied Sciences and Technology*, *5*(10). https://doi.org/10.33564/IJEAST.2021.v05i10.046

Sikandar, T., Ghazali, K. H., & Rabbi, M. F. (2019). ATM crime detection using image processing integrated video surveillance: A systematic review. *Multimedia Systems*, *25*(3), 229-251. https://doi.org/10.1007/s00530-018-0599-4

Sun, H.-M., Chen, S.-T., Yeh, J.-H., & Cheng, C.-Y. (2018). A Shoulder Surfing Resistant Graphical Authentication System. *IEEE Transactions on Dependable and Secure Computing*, *15*(2), 180-193. https://doi.org/10.1109/TDSC.2016.2539942

Syahputra, V. (2023). *Winotify* [Python]. https://github.com/versa-syahptr/winotify (Obra original publicada en 2021)

Toor, A. S., Wechsler, H., Nappi, M., & Choo, K.-K. R. (2018). Visual Question Authentication Protocol (VQAP). *Computers & Security*, *76*, 285-294. https://doi.org/10.1016/j.cose.2017.11.017

Top 23 face-recognition Open-Source Projects (Aug 2023). (s. f.). Recuperado 14 de agosto de 2023, de https://www.libhunt.com/topic/face-recognition

Tu, X., Zhao, J., Xie, M., Du, G., Zhang, H., Li, J., Ma, Z., & Feng, J. (2019). *Learning Generalizable and Identity-Discriminative Representations for Face Anti-Spoofing* (arXiv:1901.05602). arXiv. https://doi.org/10.48550/arXiv.1901.05602

Vega Briceño, E. (2021). *Seguridad de la información* (1.ª ed.). Editorial Científica 3Ciencias. https://doi.org/10.17993/tics.2021.4

Wang, C., Wang, Y., Chen, Y., Liu, H., & Liu, J. (2020). User authentication on mobile devices: Approaches, threats and trends. *Computer Networks*, *170*, 107118. https://doi.org/10.1016/j.comnet.2020.107118

Wang, Z., Xu, Y., Wu, L., Han, H., Ma, Y., & Li, Z. (2023). Improving Face Anti-spoofing via Advanced Multi-perspective Feature Learning. *ACM Transactions on Multimedia Computing, Communications, and Applications*, *19*(6), 212:1-212:18. https://doi.org/10.1145/3575660

Welcome to Face Recognition's documentation! —Face Recognition 1.4.0 documentation. (s. f.). Recuperado 14 de agosto de 2023, de https://face-recognition.readthedocs.io/en/latest/index.html

Why Does Scrum Work? 6 Reasons Why & Key Benefits of Scrum. (s. f.). Agilest®. Recuperado 21 de agosto de 2023, de https://www.agilest.org/scrum/why-does-scrum-work/

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., … Rush, A. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38-45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Zhou, L., Wang, K., Lai, J., & Zhang, D. (2023). A Comparison of a Touch-Gesture- and a Keystroke-Based Password Method: Toward Shoulder-Surfing Resistant Mobile User Authentication. *IEEE Transactions on Human-Machine Systems*, *53*(2), 303-314. https://doi.org/10.1109/THMS.2023.3236328

Zhu, Z., & Cheng, Y. (2020). Application of attitude tracking algorithm for face recognition based on OpenCV in the intelligent door lock. *Computer Communications*, *154*, 390-397. https://doi.org/10.1016/j.comcom.2020.02.003

Saabith, A. S., Fareez, M. M. M., & Vinothraj, T. (2019). Python current trend applications-an overview. International Journal of Advance Engineering and Research Development, 6(10)

Davis E. King. Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research, 2009

Nielsen, J. (2005). Ten usability heuristics

Ponemon Institute. 2016. Global Visual Hacking Experimental Study: Analysis. (2016)