



**UNIVERSIDAD DE GUAYAQUIL  
FACULTAD DE INGENIERÍA INDUSTRIAL  
CARRERA DE INGENIERIA EN TELEINFORMÁTICA**

**TRABAJO DE TITULACIÓN  
PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN TELEINFORMÁTICA**

**ÁREA  
INTELIGENCIA ARTIFICIAL**

**TEMA  
“DISEÑO API DE RECONOCIMIENTO FACIAL  
IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE  
ASISTENCIA A ESTUDIANTES”**

**AUTOR  
CHILAN PINCAY ANTHONY LIMBER**

**DIRECTORA DEL TRABAJO  
ING. SIST. GARCÍA TORRES INGRID ANGÉLICA, MG.**

**GUAYAQUIL, JULIO 2020**



**ANEXO XI.- FICHA DE REGISTRO DE TRABAJO  
DE TITULACIÓN  
FACULTAD DE INGENIERÍA INDUSTRIAL  
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



REPOSITORIONACIONAL EN CIENCIA Y TECNOLOGÍA			
FICHA DE REGISTRO DE TRABAJO DE TITULACIÓN			
TÍTULO Y SUBTÍTULO:			
DISEÑO API DE RECONOCIMIENTO FACIAL IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE ASISTENCIA A ESTUDIANTES.			
AUTOR(ES) (apellidos/nombres):		Chilán Pincay Anthony Limber	
REVISOR(ES)/TUTOR(ES) (apellidos/nombres):		Ing. Acosta Guzmán Leonel Iván / Ing. García Torres Ingrid Angélica.	
INSTITUCIÓN:		Universidad de Guayaquil	
UNIDAD/FACULTAD:		Facultad Ingeniería Industrial	
MAESTRÍA/ESPECIALIDAD:			
GRADO OBTENIDO:		Ingeniería en Teleinformática	
FECHA DE PUBLICACIÓN:	21/10/2020	No. DE PÁGINAS:	89
ÁREAS TEMÁTICAS:		Inteligencia Artificial	
PALABRAS CLAVES/ KEYWORDS:		Reconocimiento Facial, Tensor Flow, Detección de rostros, Visión Artificial, OpenCv.JS.	
RESUMEN/ABSTRACT (100-150 palabras):			
<p>El control de asistencia a estudiantes de instituciones educativas se toma de forma periódica y a través de un proceso manual, para superar esta situación se realizó la presente investigación que incluye la revisión de tecnologías de reconocimiento facial para diseñar una API que se encargue del registro y control de asistencia de los estudiantes, que pueda ser utilizado en múltiples sistemas, se utilizó la metodología de investigación bibliográfica y cuantitativa, empleando el instrumento encuesta a docentes</p>			

de la Universidad Estatal del Sur de Manabí evaluando el interés de esa población en el diseño propuesto, se realizó la revisión de tecnologías previamente utilizadas para diseño de API, se investigó la evolución de software empleados en servicios a fines, se empleó el uso librerías gratuitas como OpenCv y NodeJs. Lográndose con ella la construcción del diseño para el uso del control de asistencia acoplable a sistemas actuales y con tecnología moderna.

The fundamental base in the elaboration of this thesis, is to make an API design with facial recognition for the control of attendance to students, that means by the process of detection, registry saving and control of students in virtual or presential platforms and its scalability, facility of implementation and communication with multiple systems, as it is described next: In the first part, the problem of the teacher's attendance to his students is evaluated. In the second part, the definitions, background on the use and functionality of the API design are mentioned, as well as the process and face detection through the free web library implemented by OpenCv. Finally, the methodology, operation, work environment, software resources and benefits provided by the API for attendance control through face detection, and implementation with other systems are evaluated.

ADJUNTO PDF:	SI	X	NO
CONTACTO CON AUTOR/ES:	Teléfono: 0999482412		E-mail: anthony.chilanp@ug.edu.ec
CONTACTO CON LA INSTITUCIÓN:	Nombre: Ing. Ramón Maquilón Nicola		
	Teléfono: 593-2658128		
	E-mail: direccionti@ug.edu.ec		



**ANEXO XII.- DECLARACIÓN DE AUTORÍA Y DE  
AUTORIZACIÓN DE LICENCIA GRATUITA  
INTRANSFERIBLE Y NO EXCLUSIVA PARA EL USO NO  
COMERCIAL DE LA OBRA CON FINES NO ACADÉMICOS**



**FACULTAD DE INGENIERÍA INDUSTRIAL  
CARRERA INGENIERÍA EN TELEINFORMÁTICA**

---

LICENCIA GRATUITA INTRANSFERIBLE Y NO COMERCIAL DE LA OBRA CON  
FINES NO ACADÉMICOS

Yo, **CHILAN PINCAY ANTHONY LIMBER**, con C.C. No. **1316262193**, certifico que los contenidos desarrollados en este trabajo de titulación, cuyo título es “**DISEÑO API DE RECONOCIMIENTO FACIAL IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE ASISTENCIA A ESTUDIANTES**” son de mi absoluta propiedad y responsabilidad, en conformidad al Artículo 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, autorizo la utilización de una licencia gratuita intransferible, para el uso no comercial de la presente obra a favor de la Universidad de Guayaquil.

A handwritten signature in blue ink, appearing to read "Chilan Pincay Anthony Limber", written over a horizontal line.

**CHILAN PINCAY ANTHONY LIMBER**  
**C.C. No. 1316262193**



## ANEXO VII.- CERTIFICADO PORCENTAJE DE SIMILITUD

### FACULTAD DE INGENIERÍA INDUSTRIAL CARRERA INGENIERÍA EN TELEINFORMÁTICA



Habiendo sido nombrado ING. GARCÍA TORRES INGRID ANGELICA, tutor del trabajo de titulación certifico que el presente trabajo de titulación ha sido elaborado por **CHILAN PINCAY ANTHONY LIMBER**, con mi respectiva supervisión como requerimiento parcial para la obtención del título de INGENIERO EN TELEINFORMÁTICA.

Se informa que el trabajo de titulación: “**DISEÑO API DE RECONOCIMIENTO FACIAL IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE ASISTENCIA A ESTUDIANTES**”, ha sido orientado durante todo el periodo de ejecución en el programa Antiplagio (URKUND) quedando el 4% de coincidencia.

**URKUND**

Documento	<a href="#">tesisAchilan.docx</a> (D80456337)
Presentado	2020-10-01 20:11 (-05:00)
Presentado por	anthony.chilanp@gmail.com
Recibido	ingrid.garcia.ug@analysis.urkund.com
Mensaje	<a href="#">Mostrar el mensaje completo</a>

4% de estas 20 páginas, se componen de texto presente en 5 fuentes.

Lista de fuentes Bloques

+	Categoría	Enlace/nombre de archivo	-
+		<a href="https://siccascatastro.blogspot.com/2019/08/?m=1">https://siccascatastro.blogspot.com/2019/08/?m=1</a>	✓
+		<a href="#">PLUAS LINDAO DORA.docx</a>	-
+		<a href="#">PLUAS LINDAO DORA - URKUND.docx</a>	-
+	>	T-EBRIONES_V5 - copia.docx	✓
+		<a href="#">Tesis Portilla.docx</a>	✓
+	Fuentes alternativas		

**ING. GARCÍA TORRES INGRID ANGELICA, MG**  
**DOCENTE TUTOR**  
**CC: 1308497682**



**ANEXO VI. - CERTIFICADO DEL DOCENTE-TUTOR DEL  
TRABAJO DE TITULACIÓN**

**FACULTAD DE INGENIERÍA INDUSTRIAL  
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



Guayaquil, 30 de septiembre del 2020.

Sr (a).

**Ing. Annabelle Lizarzaburu Mora, MG.**

Director (a) de Carrera Ingeniería en Teleinformática / Telemática

**FACULTAD DE INGENIERÍA INDUSTRIAL DE LA UNIVERSIDAD DE  
GUAYAQUIL**

Ciudad. -

De mis consideraciones:

Envío a Ud. el Informe correspondiente a la tutoría realizada al Trabajo de Titulación **“DISEÑO API DE RECONOCIMIENTO FACIAL IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE ASISTENCIA A ESTUDIANTES”** del estudiante **CHILAN PINCAY ANTHONY LIMBER**, indicando que ha (cumplido con todos los parámetros establecidos en la normativa vigente:

- El trabajo es el resultado de una investigación.
- El estudiante demuestra conocimiento profesional integral.
- El trabajo presenta una propuesta en el área de conocimiento.
- El nivel de argumentación es coherente con el campo de conocimiento.

Adicionalmente, se adjunta el certificado de porcentaje de similitud y la valoración del trabajo de titulación con la respectiva calificación.

Dando por concluida esta tutoría de trabajo de titulación, **CERTIFICO**, para los fines pertinentes, que la estudiante está apta para continuar con el proceso de revisión final.

Atentamente,

**Ing. Ingrid García Torres, MG.**  
C.C. 1308497682

FECHA: 30 de septiembre del 2020



**ANEXO VIII.- INFORME DEL DOCENTE REVISOR  
FACULTAD DE INGENIERÍA INDUSTRIAL  
CARRERA INGENIERÍA EN TELEINFORMÁTICA**



Guayaquil, 15 de octubre de 2020.

Sr (a).

**Ing. Annabelle Lizarzaburu Mora, MG.**

Director (a) de Carrera Ingeniería en Teleinformática / Telemática

**FACULTAD DE INGENIERÍA INDUSTRIAL DE LA UNIVERSIDAD DE GUAYAQUIL**

Ciudad. -

De mis consideraciones:

Envío a Ud. el informe correspondiente a la REVISIÓN FINAL del Trabajo de Titulación **“DISEÑO API DE RECONOCIMIENTO FACIAL IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE ASISTENCIA A ESTUDIANTES”** del estudiante **CHILAN PINCAY ANTHONY LIMBER**. Las gestiones realizadas me permiten indicar que el trabajo fue revisado considerando todos los parámetros establecidos en las normativas vigentes, en el cumplimiento de los siguientes aspectos:

Cumplimiento de requisitos de forma:

El título tiene un máximo de 14 palabras.

La memoria escrita se ajusta a la estructura establecida.

El documento se ajusta a las normas de escritura científica seleccionadas por la Facultad.

La investigación es pertinente con la línea y sublíneas de investigación de la carrera.

Los soportes teóricos son de máximo 5 años.

La propuesta presentada es pertinente.

Cumplimiento con el Reglamento de Régimen Académico:

El trabajo es el resultado de una investigación.

El estudiante demuestra conocimiento profesional integral.

El trabajo presenta una propuesta en el área de conocimiento.

El nivel de argumentación es coherente con el campo de conocimiento.

Adicionalmente, se indica que fue revisado, el certificado de porcentaje de similitud, la valoración del tutor, así como de las páginas preliminares solicitadas, lo cual indica el que el trabajo de investigación cumple con los requisitos exigidos.

Una vez concluida esta revisión, considero que el estudiante está apto para continuar el proceso de titulación. Particular que comunicamos a usted para los fines pertinentes.

Atentamente,



Firmado electrónicamente por.

**IVAN LEONEL  
ACOSTA GUZMAN**

ING. IVÁN LEONEL ACOSTA GUZMÁN, MG

C.C:1308497682

FECHA: 14 de Octubre de 2020

### **Dedicatoria**

A mis padres Limber Chilán y Gloria Pincay por todo su apoyo durante esta época de mi vida que me impulsaron a seguir este camino y cumplir mis metas.

A mis amigos por su apoyo incondicional y haber compartido momentos gratos y difíciles en el transcurso del estudio universitario.



## **Agradecimiento**

A mis amigos que incondicionalmente me brindaron todo su apoyo durante mis años de estudio en esta institución.

A la Ingeniera Ingrid García por todas sus enseñanzas, consejos y apoyo, antes y durante el desarrollo de esta tesis.

A mis padres que me brindaron todo su apoyo para lograr cumplir mis metas, a mi hermana Adriana Chilán que es mi ejemplo a seguir.

A mis docentes, agradecer por todas sus enseñanzas impartidas y consejos durante mi proceso de estudio, por su dedicación y compromiso para mi formación profesional.

## Índice General

N°	Descripción	Pág.
	Introducción	1

### Capítulo I

#### El problema

N°	Descripción	Pág.
1.2	Objeto de la investigación	3
1.3	Sistematización del problema	4
1.4	Determinación del tema	4
1.5	Objetivos	4
1.5.1	Objetivo general	4
1.5.2	Objetivos específicos	4
1.6	Justificación	4
1.7	Delimitación del problema	5

### Capítulo II

#### Marco Teórico

N°	Descripción	Pág.
2.1	Antecedentes de la investigación.	6
2.2	Reconocimiento facial.	9
2.2.2	Proceso detección facial	9
2.2.2	Algoritmo de reconocimientos de caras	9
2.2.3	Proceso detección de rostros	10
2.2.4	Métodos de reconocimiento facial	12
2.2.4.1	Fisherfaces	12
2.2.4.2	Eigenfaces	14
2.2.5	Componentes API	15
2.2.6	Diseño API	16
2.2.7	Ventajas API REST	17

<b>N°</b>	<b>Descripción</b>	<b>Pág.</b>
2.2.8	Desventajas API REST.	17
2.2.9	Anatomía API REST.	18
2.2.10	Python	19
2.3	Marco conceptual	20
2.3.1	Adaboost	20
2.3.2	Api	21
2.3.3	Api REST	21
2.3.4	NodeJs.	22
2.3.5	OpenCv en javascript.	22
2.3.6	Web assembly	23
2.3.7	Emscripten	23
2.3.8	Base de datos.	23
2.4	Definiciones conceptuales	23
2.5	Marco contextual.	25
2.5.1	Demostración de desarrollo API.	25
2.6	Marco legal	29
2.6.1	Artículos investigativos	29

### **Capítulo III**

#### **Metodología y Propuesta**

<b>N°</b>	<b>Descripción</b>	<b>Pág.</b>
3.1	Modalidad de la investigación	45
3.2	Enfoque de la investigación	45
3.2.1	Investigación técnica	47
3.2.2	Investigación cuantitativa	47
3.2.3	Investigación cualitativa	47
3.3	Población	47

<b>N°</b>	<b>Descripción</b>	<b>Pág.</b>
3.3.1	Ejecución de encuestas	49
3.3.2	Análisis de resultados del modelo de encuesta	49
3.3.3	Resultados de la encuesta	49
3.3.4	Análisis de la encuesta	44
3.4	Propuesta de investigación	56
3.5	Esquema general de simulación	57
3.6	Servidor XAMPP	57
3.7	Server NodeJs	58
3.8	Estructura base de datos	60
3.8.1	Detalle de la base de datos	61
3.8.2	Modelo de la base de datos	61
3.8.3	Información de ambiente de programación	62
3.9	Creación de API de reconocimiento facial	62
3.10	Resultados	63
3.10.1	Descripción de funcionalidades	63
3.10.2	Reconocimiento facial único por imagen	63
3.10.3	Agregar usuarios	64
3.10.4	Extracción de rostros	64
3.11	Conclusiones	66
3.12	Recomendaciones	66
	<b>Anexos</b>	<b>67</b>
	<b>Bibliografía</b>	<b>87</b>

## Índice de tablas

<b>N°</b>	<b>Descripción</b>	<b>Pág.</b>
1	Desventajas y ventajas de reconocimiento facial	22
2	Artículos investigativos	40
3	Uso de software libre en el Ecuador	40
4	Constitución de la república del Ecuador	41
5	Régimen del buen vivir	41
6	Ley Orgánica de educación superior	41
7	Código orgánico penal	42
8	Variable de investigación	43
9	Recurso de software	44
10	Recurso de hardware - software	45
11	Métodos de toma de asistencia	47
12	Proceso de detección de rostros	48
13	Visión artificial para proceso de detección de rostros	49
14	Aplicación de reconocimiento de rostros	50
15	Uso API para detección de rostros	51
16	Herramienta de rostros para el control de asistencia	52
17	Tipos de tecnología para el control de asistencia	53
18	Uso API de reconocimiento facial	54
19	Descripción de la base de datos	60
20	Estructura del ambiente de programación	62

## Índice de figuras

<b>N °</b>	<b>Descripción</b>	<b>Pág.</b>
1	Proceso de reconocimiento facial	9
2	Proceso de cascada con imágenes positivas y negativas	10
3	Resultado de aplicar Fisher faces de Revista colombiana de tecnologías de avanzada	14
4	Diagrama del diseño API.	17
5	Software python	19
6	Representación adaboost	20
7	Proceso de levantamiento de server MySQL con Nodejs	25
8	Formulario Ingreso de información de usuarios	26
9	Subida de datos de personas mediante el explorador de Windows	26
10	Extracción de rostros por NodeJs	27
11	Repuesta a usuarios reconocidos	28
12	Respuesta personas mediante el navegador	28
13	Métodos de toma de asistencia a estudiantes	37
14	Proceso de detección de rostros	38
15	Visual artificial para proceso de detección de rostros	39
16	Aplicación de reconocimiento de rostros	40
17	Uso API para detección de rostros	41
18	Herramienta de rostros para el control de asistencia	42
19	Tipos de tecnología para el control de asistencia	43
20	Uso API de reconocimiento facial	44
21	Estructura Backend	46
22	Estructura Frontend.	46
23	Levantamiento de Servidor MySQL	47
25	Creación de base de datos	47
26	IDE de desarrollo visual studio code	48
27	Instalación MySQL en NodeJs	50
28	Modelo de Base de Datos en dbForge Studio	51
29	Sidebar en sistema api de reconocimiento facial	52

<b>N °</b>	<b>Descripción</b>	<b>Pág.</b>
29	Sidebar en sistema api de reconocimiento facial	52
31	Rostros reconocidos por funcionalidad de imagen	54
32	Formulario de registro de usuarios	55
33	Extracción de rostros reconocidos API	56
34	Salida de personas detectadas en formato JSON	56

## Índice de anexos

<b>N °</b>	<b>Descripción</b>	<b>Pág.</b>
1	Manual NodeJs	68
2	Artículos Investigativos	70
3	Uso de software libre del Ecuador	71
4	Constitución de la república del Ecuador	72
5	Régimen del buen vivir	73
6	Ley Orgánica de Educación Superior	74
7	Código orgánico integral penal	75
8	Modelo de la encuesta	76
9	Instalación de XAMPP	78
10	Conexión con la base de datos mediante NodeJs	81
11	Levantamiento de servidor en puerto 3000	82
12	Instalación de dbForge Studio	84





## **ANEXO XIII.- RESUMEN DEL TRABAJO DE TITULACIÓN (ESPAÑOL)**



### **FACULTAD DE INGENIERÍA INDUSTRIAL CARRERA INGENIERÍA EN TELEINFORMÁTICA**

---

#### **“DISEÑO API DE RECONOCIMIENTO FACIAL IMPLEMENTANDO OPENCV.JS PARA EL CONTROL DE ASISTENCIA A ESTUDIANTES”**

**Autor:** Chilán Pincay Anthony Limber

**Tutor:** Ing. García Torres Ingrid Angélica, MG

#### **Resumen**

El control de asistencia a estudiantes de instituciones educativas se toma de forma periódica y a través de un proceso manual, para superar esta situación se realizó la presente investigación que incluye la revisión de tecnologías de reconocimiento facial para diseñar una API que se encargue del registro y control de asistencia de los estudiantes, que pueda ser utilizado en múltiples sistemas, se utilizó la metodología de investigación bibliográfica y cuantitativa, empleando el instrumento encuesta a docentes de la Universidad Estatal del Sur de Manabí evaluando el interés de esa población en el diseño propuesto, se realizó la revisión de tecnologías previamente utilizadas para diseño de API, se investigó la evolución de software empleados en servicios a fines, se empleó el uso librerías gratuitas como OpenCv y NodeJs. Lográndose con ella la construcción del diseño para el uso del control de asistencia acoplable a sistemas actuales y con tecnología moderna.

**Palabras claves:** Reconocimiento Facial, Tensor Flow, Detección de rostros, Visión Artificial, OpenCv.JS



## ANEXO XIV.- RESUMEN DEL TRABAJO DE TITULACIÓN (INGLÉS)

FACULTAD DE INGENIERÍA INDUSTRIAL  
CARRERA INGENIERÍA EN TELEINFORMÁTICA



---

### “API DESIGN OF FACIAL RECOGNITION IMPLEMENTING OPENCV.JS FOR STUDENT ATTENDANCE CONTROL”

**Author:** Chilán Pincay Anthony Limber

**Advisor:** Ing. García Torres Ingrid Angélica, MG

#### **Abstract**

The control of attendance to students of Educational Institutions is taken periodically through a manual process. In order to overcome this situation the present investigation was carried out on including the review of facial recognition technologies to design an API that is responsible for the registration and control of attendance that can be used in multiple systems, the Bibliographic and Quantitative research methodology was used by using the survey instrument for teachers of the Universidad Estatal del Sur de Manabí, from Jipijapa in Manabí Province, It was evaluated for the interest of this population in the proposed design, the technology review was previously carried out and also used for API design, the evolution of software in services was investigated, On the other hand, The use of free libraries such as OpenCV and NodeJs were applied.

The construction of the design for the use of acceptable attendance control to current systems and with modern technology were developed.

**Keywords** Face Recognition, Tensor Flow, Face Detection, Machine Vision, OpenCv.JS

## Introducción

Computer Vision, que generalmente se abrevia como CV, se caracteriza por ser un campo muy amplio de la inteligencia artificial que busca crear métodos para ayudar a las computadoras a "ver" y obtener la sustancia de imágenes avanzadas como fotos y videos. El objetivo de Computer Vision es extraer datos valiosos de las imágenes.

Esto ha demostrado lo sorprendentemente desafiante que es; ha poseído miles de mentes brillantes e inventivas durante las últimas cuatro décadas, y a pesar de esto todavía se está lejos de poder construir una "máquina de ver" de propósito general.

El reconocimiento facial es un método para identificar o verificar la identidad de un individuo mediante la detección de patrones faciales. Los sistemas de reconocimiento facial se pueden utilizar para identificar personas en fotos, videos o en tiempo real. La policía también puede usar dispositivos móviles para identificar personas durante las paradas policiales.

Pero los datos de reconocimiento facial pueden ser propensos a errores, lo que puede implicar a las personas por delitos que no han cometido. El software de reconocimiento facial presenta falencias para reconocer a los afroamericanos y otras minorías étnicas, mujeres y jóvenes, a menudo identificándolos erróneamente o no identificándolos, impactando de manera dispar a ciertos grupos (E3FF, 2014).

El reconocimiento facial abarca muchos ámbitos de tecnología incluso con los controles de acceso como los biométricos, y son adaptables a cualquier sistema pudiendo ser web en cualquier sistema operativo,

Los Biométricos se utiliza para la seguridad y / o autenticación de transacciones. Un sistema biométrico es una forma efectiva de verificar la identidad de una persona. Se utilizan en aeropuertos, investigaciones forenses, edificios seguros y gestión de la fuerza laboral. Tomando que las unidades administrativas han obtenido beneficio de este tipo de tecnología . En este trabajo se propone el diseño de un api usando inteligencia artificial mediante aprendizaje por tareas como Deep Learning y derivados. Debido a esto se usará un sistema biométrico sencillo y se diseñará un sistema de reconocimiento capaz de aprender por sí mismo, con una auto eficiencia y a bajo coste.

Este diseño utilizará lenguajes de programación de alto nivel, con lado al cliente usando datos y requerimientos de manera asíncrona como lo es nodejs y usando una librería de

computer visión como lo es OPENCV.JS que ayudara con la forma y reconocimiento del rostro de una manera eficiente. El presente trabajo consta de 3 etapas:

Etapa 1: El diseño del backend como etapa 1.

Etapa 2: Uso de NodeJs y una base de datos robusta como lo es MySQL para un mejor control y flujo de datos, como intermedio entre el backend y frontend.

Etapa 3: Uso de un framework express basado en NodeJs para un diseño de interfaces más intuitiva a nivel de usuario.

# **Capítulo I**

## **El Problema**

### **1.1 Planteamiento del problema.**

La revisión de asistencia de personas no ha ido evolucionando en medida de los avances tecnológicos que se ve en el presente. El desarrollo tecnológico actual busca automatizar procesos que llevan a cabo cualquier empresa pública o privada con el fin de mejorar su ambiente laboral y personal. Es muy habitual que los docentes que aun usen programas como Excel o aplicativos similares para el control de asistencia de estudiantes, o documentos que firma a diario para el control mismo.

Reconocimiento facial es el sistema de control de acceso biométrico que se impone en este momento para el control de accesos por sobre el lector de huella digital dactilar. Es más eficaz y seguro, ya que no requiere contacto físico con el beneficiario y por lo tanto evita la transmisión de agentes contaminantes e infecciones, conjuntamente de permitir medir la temperatura de los usuarios y si estos están utilizando mascarilla de protección, cuestión estrechamente trascendental y relevante en estos momentos para cualquier empresa y lugar de tráfico masivo, en momentos de la pandemia mundial COVID-19. Las APIs permiten que sean los equipos informáticos y no las personas las que manejen el trabajo. A través de las APIs las personas pueden actualizar los flujos de trabajo para que se puedan realizar en menos tiempo a la vez que sean más productivos. Las APIs al consistir en códigos predefinidos en otros softwares, ahorran trabajo y son mucho más rentables, ya que no es necesario inventar lo que ya está inventado. Además, son códigos probados que sin duda serán funcionales y seguros. El presente proyecto está enfocado en el diseño de un api de reconocimiento facial que pueda comunicarse en cualquier sistema de control de asistencia a estudiantes. Existen diferentes alternativas de sistemas que tienen reconocimiento facial, pero ninguna que pueda conectarse con otros sistemas y ser integrado como un api tanto como con un iframe o por webservice para ser consumido desde cualquier sistema compatibles.

### **1.2 Objeto de la investigación**

Mediante el diseño de un API que integre reconocimiento facial para el control de acceso, comprobar las condiciones en la que beneficia este tipo de sistemas implementando esta tecnología. Esto comprobará la posible solución a la problemática en la toma de asistencia en términos de portabilidad, control y acceso a un lugar determinado.

### 1.3 Sistematización del problema

- ¿Qué métodos de toma de asistencia a estudiantes existen actualmente?
- ¿Qué posibilidad existen de implementar en sistemas actuales de control de asistencia?
- ¿Qué mejora podría usarse para la toma de asistencias a estudiantes?
- ¿Qué tipo de tecnologías existen para el proceso de toma de asistencia a estudiantes?

### 1.4 Determinación del tema

“Diseño API de reconocimiento facial implementando OpenCV.js para el control de asistencia a estudiantes.”

### 1.5 Objetivos

#### 1.5.1 Objetivo General

Diseñar un API de reconocimiento facial usando opencv.js para el control de asistencia a estudiantes.

#### 1.5.2 Objetivos Específicos

- Revisar las tecnologías empleadas para el proceso de reconocimiento facial.
- Determinar los recursos de software y hardware requeridos para realizar el diseño de la solución del control de asistencia automatizada.
- Realizar el diseño de un API orientada al servicio de control de asistencia.

### 1.6 Justificación

Se considera que conforme pasa el tiempo en las ramas tecnológicas como la inteligencia artificial se implementa en muchos ámbitos actuales como la salud, comercio, educación, y servicios financieros.

El desarrollo del sistema de control de asistencia con reconocimiento facial tiene como finalidad la colaboración directa con el sistema educativo, específicamente hacia los docentes que en la actualidad utilizan plataformas de gestión de aprendizaje, para impartir clases, garantizando la eficiencia, fácil manejo y la identificación rápida de los participantes de la clase, ya que la abstracción de datos será mediante imágenes o en tiempo real por medio de la videocámara que podrán llevarla a cabo durante la sesión.

La propuesta del diseño del sistema basado en Ux (User Experience) tendrá como características la portabilidad, escalabilidad, y seguridad al momento de llevar el registro. Surge la necesidad de la creación del sistema inteligente con el fin de facilitar la tarea del docente,

acogerse a la tecnología dejando a un lado los registros mediante documentación, con un manejo más intuitivo para cada usuario con mejor facilidad de acceso.

Para ello mediante el análisis de las herramientas que se ajusten mejor al diseño del sistema se logrará la determinación de la demanda, eficiencia, para que la implementación de la API se convierta en un apoyo para el sistema de control asistencia a estudiantes.

### **1.7 Delimitación del problema**

Los procesos rutinarios han tenido gran afectación debido a la emergencia sanitaria, la propagación de las enfermedades virales que provocan pandemias mundiales ha vuelto a centrar la atención en la higiene y la salud. El sector de la salud, la economía del país y el sistema educativo han sufrido cambios en la actualidad, lo que ha llevado al uso continuo de la tecnología para facilitar las tareas cotidianas en las familias ecuatorianas.

Actualmente en el sistema educativo, los docentes se han visto en la obligación de impartir clases por medio de plataformas de enseñanza virtuales, lo que en ocasiones se les dificulta llevar un control de la asistencia, de forma eficaz y dar seguimiento de faltas a las clases de los estudiantes de forma periódica, esta problemática conlleva a la falta de herramientas adecuadas para la gestión académica y administrativa de los centros educativos.

Simplificar el proceso rutinario por una rutina de procesos creados con API que implementa OPENCV.JS en un sistema web que se puede comunicar con cualquier sistema es una parte importante en beneficio de controlar estos procesos.

## **Capítulo II**

### **Marco Teórico**

#### **2.1 Antecedentes de la investigación.**

Existen varios tipos de sistemas en distintas plataformas que usan el reconocimiento facial como una parte importante para el control de asistencia a estudiantes. Hace mucho tiempo la digitalización de estos procesos es una de las bases importantes para los sistemas de identificación, actualmente estos desarrollos han aportado en la identificación de las personas, el acceso a sitios privados y en todos los lugares que necesiten seguridad y orden con la ayuda de ordenadores computarizados que realizan el reconocimiento facial, exclusivamente basado en imágenes de rostros humanos para su función. El proceso de detección facial hace uso de algoritmos para determinar si hay rostros humanos en una foto o video, es decir, solo determina si hay alguna cara, mas no a quien pertenece.

La Universidad Central del Ecuador se planteó desarrollar un Sistema Computarizado para llevar el registro y control de asistencia de los empleados de la Facultad de Ingeniería, con la finalidad de optimizar las funciones y aumentar la confiabilidad de la organización con trabajadores, para mejorar el manejo de los horarios de los empleados, “Se estudió el funcionamiento del sistema actual y se descubrió que posee debilidades, los requerimientos de la información por parte del personal son mínimas, no existen los reportes requeridos, registros de inasistencia, de atrasos, ingreso y administración de los diferentes horarios de cada empleado y otras deficiencias, por lo que se hizo necesario proponer e implementar un nuevo sistema computarizado para el control de personal acorde con el desarrollo de la tecnología” (Maza & Goyes Mosquera, 2015).

El proceso de integración de reconocimiento facial es uno de los problemas más fundamentales en los sistemas, ya que para usar el servicio se requieren programas externos que no se integran al sistema que tienen. Por esa misma razón es que existen paquetes de sistemas que usan esta tecnología, pero sin comunicarse entre sí. Un API ha buscado la solución a programadores al momento de empezar sin bases del desarrollo de algún sistema de software, usando rutas como parámetro principal en la que se obtuvo un método previamente creado para la comunicación entre funciones de dichos sistemas.



## 2.2 Reconocimiento Facial.

Los sistemas de reconocimiento facial son una tecnología del campo de la inteligencia artificial que permite identificar a una persona analizando las características biométricas de su rostro. En una fase inicial, se usaban modelos geométricos simples, pero actualmente el análisis está ligado a sofisticados procesos matemáticos y algoritmos de coincidencia. Estos, consiguen medir la forma y las estructuras únicas de los rostros.

“Hoy existen diferentes sistemas de reconocimiento facial que utilizan métodos distintos. Pero en la base, esta tecnología usa los mismos principios que otras técnicas de autenticación biométrica, como los escáneres de huellas digitales y el reconocimiento de voz” (Martinez, 2019).

La base en sistemas basados en el control de asistencia, control de acceso y derivados es la autenticación de personas, en la cual muestra un beneficio de seguridad en comparación a las huellas digitales o reconocimiento de voz.

**Tabla 1.** Ventajas y desventajas del reconocimiento facial

<b>Ventajas</b>	<b>Desventajas</b>
Versatilidad en la toma de asistencia a personas.	Menos seguro en la toma de datos del sistema.
Mejora en procesos manuales en la toma de asistencias	Problemas en la obtención de datos al proceso de asistencias.
Más higiene para evitar contagios.	Menor control físico y problemas al reconocer el rostro.

*Información tomada de la investigación directa. Elaborado por Chilán Pincay Anthony Limber*

### 2.2.1 Proceso Detección Facial.

El proceso de detección facial está dado por cuatro pasos: Detección de la cara, Alineación de la cara, extracción de las características para finalmente realizar el reconocimiento (ver figura 1). Se detecta que hay una cara en la imagen, sin identificación previa, determinando la escala en la que se encuentra, luego localiza los componentes de la cara, mediante transformaciones geométricas, lo que permite la alineación, acondicionamiento y normalización, se extrae las características, con la ayuda del algoritmo

de reconocimiento, finalmente el sistema procede a realizar las comparaciones con la base de datos, para luego identificar el rostro del individuo.

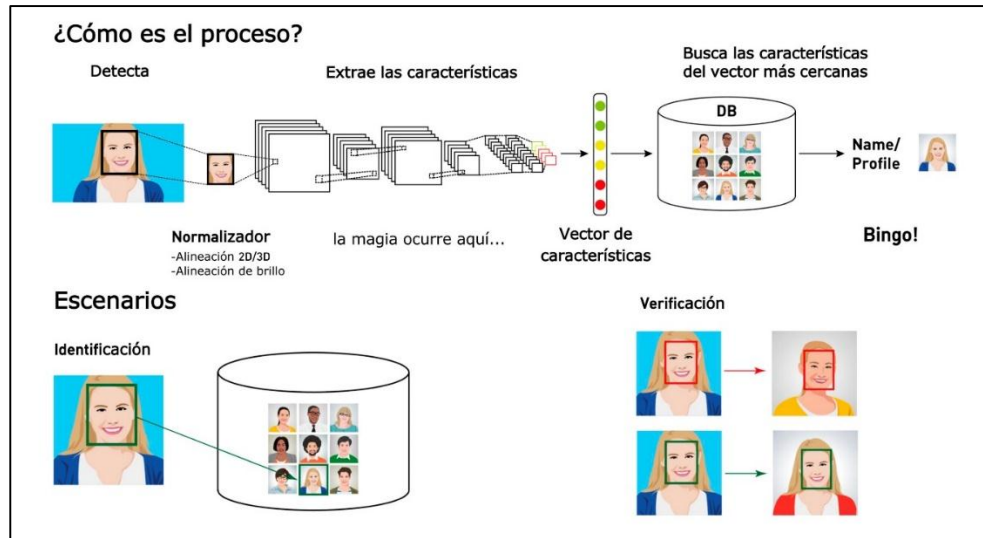


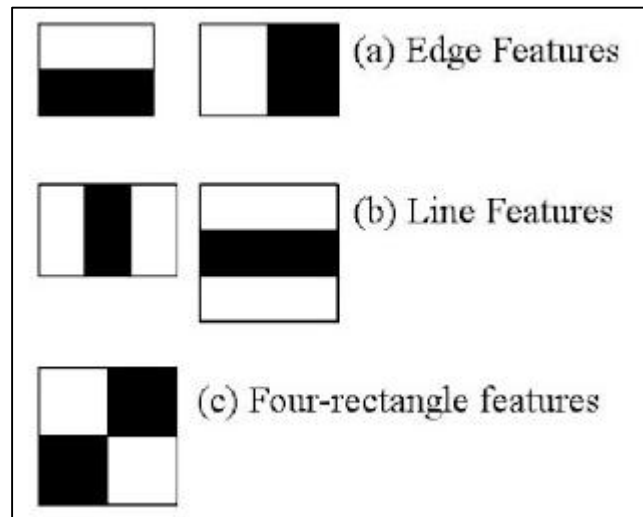
Figura 1. Proceso de reconocimiento facial, Información tomada de kimaldi.com. Tomado de la investigación previa.

### 2.2.2 Algoritmo de Reconocimientos de caras

El proceso de reconocimiento facial consiste en tomar varias imágenes de las dimensiones,  $a$  filas y  $b$  columnas, la que se transforman en un vector unitario contenido en un espacio de imágenes  $n$ -dimensionales ( $n = a \times b$ ). Posteriormente, se obtiene la imagen promedio y se la proyecta junto con el valor resultante en un subespacio de la menor dimensión utilizando uno de los métodos de reducción de dimensión como la extracción de características. Esta proyección será comprada con la proyección de un conjunto de imágenes de la base de datos previamente configurada en MySQL, la clase del vector más similar será el resultado que se muestre como el proceso de reconocimiento del sistema de asistencia. (Ottado, 2015)

### 2.2.3 Proceso Detección de Rostro.

La detección de objetos utilizando clasificadores en cascada basados en características de Haar como "Detección rápida de objetos mediante una cascada mejorada de características simples" (Viola & Jones, 2001). Es un enfoque basado en el aprendizaje automático en el que se entrena una función en cascada a partir de muchas imágenes positivas y negativas. Luego se usa para detectar objetos en otras imágenes como en la figura 2.



*Figura 2 Proceso de cascada con imágenes positivas y negativas Información obtenida de opencv.org. Tomado de la investigación previa.*

Ahora todos los tamaños y ubicaciones posibles de cada kernel se utilizan para calcular muchas características. Para el cálculo de cada característica, se necesita encontrar la suma de los píxeles debajo de los rectángulos blanco y negro. Para solucionar esto, se introdujeron las imágenes integrales. Simplifica el cálculo de la suma de píxeles, qué tan grande puede ser el número de píxeles, a una operación que involucra solo cuatro píxeles.

Pero entre todas estas características que se calcula, la mayoría son irrelevantes. Por ejemplo, la imagen a continuación. La fila superior muestra dos buenas características. La primera característica seleccionada parece centrarse en la propiedad de que la región de los ojos suele ser más oscura que la región de la nariz y las mejillas. La segunda característica seleccionada se basa en la propiedad de que los ojos son más oscuros que el puente de la nariz. Pero las mismas ventanas que se aplican en las mejillas o en cualquier otro lugar son irrelevantes. Esto se logra con Adaboost.

El clasificador final es una suma ponderada de estos clasificadores débiles. Se llama débil porque por sí solo no puede clasificar la imagen, pero junto con otras formas en un clasificador fuerte. Al funcionar incluso 200 funciones proporcionan detección con un 95% de precisión. Su configuración final tenía alrededor de 6000 funciones. (Imagine una reducción de más de 160000 funciones a 6000 funciones. Eso es una gran ganancia).

Así que se toma una imagen. Toma cada ventana de 24x24. Con 6000 funciones. Se comprueba si es cara o no.

En una imagen, la mayor parte de la región de la imagen es una región sin rostro. Por lo tanto, es mejor tener un método simple para verificar si una ventana no es una región de cara. Si no es así, deséchelo de una sola vez. No lo proceses de nuevo. En su lugar, céntrese en la región donde puede haber una cara. De esta manera, se podrá encontrar más tiempo para verificar una posible región facial.

Para ello se introdujo el concepto de Cascada de Clasificadores. En lugar de aplicar todas las 6000 características en una ventana, agrupe las características en diferentes etapas de clasificadores y se aplica una por una. (Normalmente, las primeras etapas contendrán un número muy reducido de funciones). Si una ventana falla en la primera etapa, deséchela. No considerar las funciones restantes en él. Si pasa, aplique la segunda etapa de características y continúe el proceso. La ventana que pasa por todas las etapas es una región de cara.

El detector de los autores tenía más de 6000 características con 38 etapas con 1, 10, 25, 25 y 50 características en las primeras cinco etapas. (Dos características en la imagen de arriba se obtienen realmente como las dos mejores características de Adaboost). Según los autores, en promedio, se evalúan 10 características de más de 6000 por sub-ventana.

## **2.2.4 Métodos de reconocimiento Facial**

### **2.2.4.1 *Fisherfaces***

Fisherface es una técnica de reconocimiento de rostros, que tiene en cuenta la luz y expresiones faciales. Este se encarga de clasificar y reducir las dimensiones de las caras usando el método FLD (Discriminant Lineal Fisher). En este sentido, el análisis discriminante de Fisher intenta proyectar los datos de manera que su nueva dispersión sea óptima para la clasificación. Mientras PCA busca los vectores que mejor describen los datos, LDA (Discriminant Lineal Analysis) busca los vectores que proporcionan mejor discriminación entre clases después de la proyección. ( Tarazona Ospina & MSc. Daniel, 2015)

Fisherfaces realiza un LDA, donde se busca aprovechar la información disponible, sobre la clasificación de las imágenes de entrenamiento, para buscar una proyección que maximice la separación entre imágenes de diferentes personas (o clases) y minimice la distancia entre imágenes de una misma clase, así logra concentrar las imágenes mejorando, en forma importante, la tasa de reconocimiento.

Se define la matriz de la varianza entre clase (imágenes de personas distintas)

$$S_B = \sum_{i=1}^c |X_i| (u_i - u)(u_i - u)$$

y varianza dentro de la clase (imágenes de una misma persona)

$$S_W = \sum_{i=1}^c \sum_{X_k \in X_i} (X_k - u_i)(u_k - u_i)$$

Donde:

$u_i$  = imagen promedio de la clase  $X_i$

$|X_i|$  = Número de puntos dentro de la clase  $X_i$

$u$  = Promedio de todas las imágenes

Luego en forma similar a PCA, se debe encontrar la matriz de proyección  $W \in R^{(m \times n)}$ .

Resolviendo el problema de optimización se obtiene la ecuación:

$$W_{opt} = \frac{\arg \max_w}{w} \left| \frac{W^T S_B W}{W^T S_W W} \right|$$

Donde:

$S_W$  = Minimizar

$S_B$  = Maximizar

El resultado de dicho problema es una matriz con los vectores propios de  $S_W S_B$ , para esto se debe reducir la dimensionalidad de las imágenes a no más de número de imágenes – número de clases. Luego se puede aplicar LDA para reducir el número de clases -1 elementos y agrupar las imágenes según la clase a la que pertenecen. Si la primera reducción se realiza utilizando Eigenfaces, al método se le llama Fisherfaces.

Este método se puede separar en dos partes, primero resolver las ecuaciones dadas en PCA, donde,  $S_W$  y  $S_B$  creadas a partir de las imágenes proyectadas sobre el subespacio de eigenfaces. Estos métodos han mostrado gran eficiencia y popularidad debido a la relativa simpleza. Sin embargo, la cantidad de cálculos y requerimientos de memoria son muy altos debido a las grandes dimensionalidades de las imágenes. ( Tarazona Ospina & MSc. Daniel, 2015)

$$W_{opt} = \frac{\arg \max}{w} \left| \frac{W^T \cdot W_{PCA}^T \cdot S_B \cdot W_{PCA} \cdot W}{W^T \cdot W_{PCA}^T \cdot S_W \cdot W_{PCA} \cdot W} \right|$$

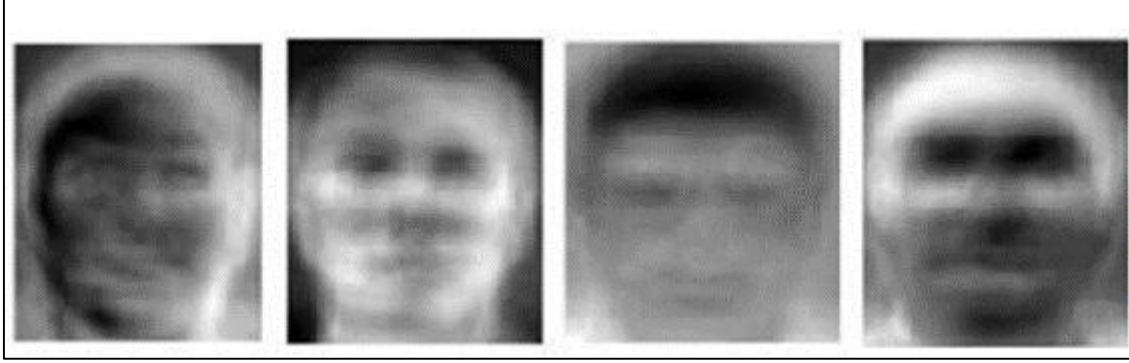


Figura 3 Resultado de aplicar Fisherfaces de Revista colombiana de tecnologías de avanzada. Elaborado por el autor.

#### 2.2.4.2 Eigenfaces

Eigenfaces realiza una proyección lineal del espacio de imágenes a un espacio de características de menor dimensión. Esta reducción se realiza utilizando la técnica de Análisis de Componentes Principales (PCA), la cual consigue una reducción de la dimensión proyectando los datos sobre la dirección que maximice la distribución total sobre todas las clases. En primer lugar, se considera un conjunto de  $N$  imágenes con valores en el espacio de imágenes  $n$ -dimensional:

$$\{X_i\} \quad i = 1, 2, \dots, N$$

Se asume además que cada una de las imágenes pertenece a una de las  $c$  clases  $\{X_1, X_2, \dots, X_c\}$ . Asimismo, se considera una transformación lineal que lleva al espacio de imágenes original de  $n$  dimensiones al espacio de características de dimensión  $m$ , donde  $m < n$ . Los nuevos vectores de características  $Y_k \in \mathbb{R}^m$  son definidos por la siguiente transformación lineal:

$$Y_k = W^T x_k \quad k = 1, 2, \dots, N$$

Donde:

$W \in \mathbb{R}^{m \times n}$  = Matriz de columnas ortogonales. Se define además la matriz de distribución total  $S_T$  como:

$$S_T = \sum_{k=1}^N (x_k - u)(x_k - u)^T$$

Donde:

$u \in R^n$  = Media de todas las imágenes de la ecuación  $\{X_i\}$   $i = 1, 2, \dots, N$ .

$W^T$  = Luego de aplicar la transformación lineal, la distribución de los vectores de características  $\{Y_1, Y_2, \dots, Y_N\}$  es  $W^T S_T W$ . Se toma aquella proyección  $W_{opt}$  que maximiza el determinante de la distribución total de la matriz de las imágenes proyectadas, esto es

$$\begin{aligned} W_{opt} &= \frac{\arg \max_w}{w} |W^T S_T W| \\ &= \{w_1, w_2, \dots, w_m\} \end{aligned}$$

Donde:

$\{w_i, |i = 1, 2, \dots, m\}$  = es el conjunto de vectores propios n-dimensionales de  $S_T$  correspondiente a los mayores  $m$  vectores propios. Dichos vectores propios tienen la misma dimensión que las imágenes originales y se les denomina eigenfaces.

### 2.2.5 Componentes API

Las funciones y procedimientos considerados en conjunto para ser utilizados por otro software se la denomina la interfaz de programación de aplicaciones, su función principal como capa de abstracción cumple un rol importante en el desarrollo de este proyecto, además de la descripción de su forma en la que los programas y los sitios webs intercambia los datos, normalmente en formato JSON (JavaScript Object Notation) o XML (Extensible Markup Language).

La funcionalidad es para ofrecer todos los datos a las aplicaciones y permite a los desarrolladores que de una forma efectiva se logre simplificar la programación, ya que el formato que posee es estándar, es decir su compatibilidad se cumple al cien por ciento y logra consumir datos de otras aplicaciones.

REST es considerado otro aporte al componente de la API, debido a que su arquitectura del desarrollo web se acopla totalmente al protocolo HTTP, la lista de reglas que en se incluyen deben cubrir a la API en todo aspecto técnico cumpliendo con el diseño de la arquitectura. REST no es un protocolo y tampoco está estandarizado, más bien la aplicabilidad consiste en una serie de principios de arquitectura, la mayoría de las peticiones son independientes, esto se interpreta como cada petición obligatoriamente debe llevar todo

el contenido para la realización, es el usuario quién se encarga de mantener el estado en la propia aplicación, en base a la información procesada por la API, permitiendo que los servidores ahorren recursos, siendo no necesario almacenar información de la sesión de usuario facilitando las labores del desarrollador de la programación.

### 2.2.6 Diseño API.

Se considera que las principales características de una API deben ser de fácil aprendizaje, utilización, evitar la documentación y apropiada para la audiencia. El proceso que se detalla a continuación es el más importante de un desarrollo de software, el diseño que consiste en mostrar la arquitectura, componentes e incluso la interfaz y otras características del sistema que se utilizará en este proyecto de titulación.

Se destaca como componente el Reconocimiento Facial, responsable de reconocer el rostro de las personas ingresadas en una clase, se lo realiza mediante imágenes, o en tiempo real por medio de la videocámara, cuando los estudiantes estén recibiendo clases virtuales en las múltiples plataformas de gestión de aprendizaje, de donde se realizará la extracción de los rostros, la cual será procesada en formato JSON, que es un medio de comunicación entre sistemas, con texto sencillo para el intercambio de datos.

El diseño del api según la comunicación con otros sistemas en la figura 1 como a continuación.

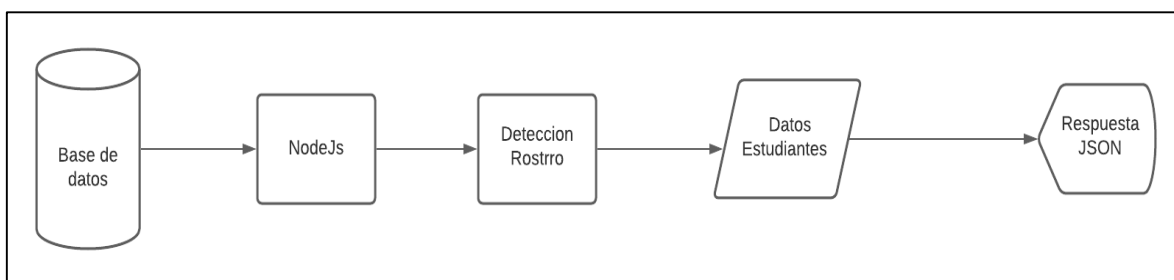


Figura 4 Diagrama del diseño API. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

### 2.2.7 Ventajas API REST.

**Agilidad o Rapidez:** El uso de las Apis de terceros permite a los desarrolladores integrar en cuestión de minutos funcionalidades como mapas, imágenes o servicios de autenticación sin la necesidad de crear desde cero estas funcionalidades y así disminuir el tiempo de desarrollo.



**Garantía de calidad:** En gran cantidad de los casos las APIs ya han sido probadas en todos los ambientes de desarrollo por lo que aseguran un óptimo desempeño en funcionalidad y en cualquiera de los ambientes que se utilice.

**Representación:** Es importante tener en cuenta de que aquellas APIs que son utilizadas por un gran número de usuario y que de una manera u otra le han brindado un buen desempeño logran crear en los usuarios un estado de seguridad al momento de utilizar dichas APIs.

**Mejora continua:** Las grandes compañías no pueden darse el lujo de permitirse ser desplazadas por la competencia así que de manera continua estas se mantienen creando nuevas funcionalidades y mejoras de seguridad a los servicios y funciones que ofrecen mediante el uso de las APIs.

#### **2.2.8 Desventajas API REST.**

**Cambios:** Se debe tener en consideración que al estar utilizando servicios de tercero en el desarrollo se debe llegar a encontrar en ocasiones que las rutinas para la implementación o utilización de la API se hayan modificado ya sea por mejora de la misma o por cuestiones de seguridad y si no está al tanto de estos cambios poder llegar a perder el acceso a la API, también se puede llegar a presentar la situación de que al momento de adquirir el acceso a la API se realizará de manera gratuita y que un momento determinado los desarrolladores decidan que este servicio deberá ser pago.

**Control de disponibilidad:** El uso de APIs ahorra tiempo en desarrollo de las aplicaciones o funcionalidades de los sitios web, también es cierto que no deja de ser un servicio de tercero del cual se debe tener control, si por algún motivo este deja de funcionar de manera repentina, cosa que muy rara vez ocurre, no estaría bajo nuestro control la solución del mismo y en todo caso la única opción sería esperar a que los desarrolladores de la API resuelvan el problema que está afectando el desempeño esperado.

**Conocimiento:** Las desventajas que se pueden presentar al momento de querer implementar por primera vez un servicio de tercero como son las APIs es que la falta de manejo de la misma o la falta del conocimiento necesario puede representar un atraso en el desempeño de la misma ya que no todas las APIs son fáciles de implementar y no todas necesitan los mismos requerimientos para su funcionamiento.

### 2.2.9 Anatomía API REST.

“Existen distintos tipos de APIs que pueden utilizar diferentes protocolos de comunicación. Por ejemplo, el protocolo FTP (File Transfer Protocol) es un protocolo de tipo asíncrono, es decir, la comunicación entre el cliente y el servidor no ocurre en tiempo real por este motivo es un tipo de protocolo que ha sido usado en situaciones en las que no era necesaria una respuesta inmediata” (Redondo, 2018).

FTP es un protocolo que se usa en ocasiones para servicios web que se ajusten a los siguientes dos requerimientos: una comunicación asíncrona y un volumen importante de información. FTP tiene otra característica que lo hacen muy interesante y es que puede gestionar el intercambio de volúmenes grandes de información de manera muy eficiente.

SFTP es la abreviatura de Secure File Transfer Protocol (Protocolo de transferencia segura de archivos). Este protocolo permite transferir datos cifrados entre tu ordenador local y el espacio web del que dispones en tu hosting de STRATO a través de Secure Shell (SSH).

El protocolo de transferencia de archivos SFTP es independiente del sistema operativo que utilices y del tipo de conexión. Además de la posibilidad de transferir archivos a través de SFTP, también permite visualizar directorios, cambiar el nombre o limitar derechos. SFTP es el sucesor de FTP: la diferencia es una mayor seguridad, como lo revela el añadido "Seguro".

El protocolo de comunicación más usado en el caso de las APIs es HTTP (HyperText Transfer Protocol), el mismo que usa diariamente para navegar por internet desde el navegador. HTTP es un protocolo de comunicación basado en el uso de mensajes de texto para gestionar peticiones a un servidor y para las respuestas desde éste. Se trata de un protocolo que es síncrono, es decir, la respuesta a la petición ocurre inmediatamente. Tanto las peticiones (request) o las respuestas (response) siguen un vocabulario y sintaxis definida. Por ejemplo, las peticiones generalmente están basadas en verbos, que son métodos como GET o POST.

### 2.2.10 Python



*Figura 5 Software Python Información tomada de la investigación previa. Elaborado por el autor.*

“Python es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso, páginas web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, lo que ofrece ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad”. (Alvarez, 2015) Python el lenguaje más popular actualmente tiene la facilidad de ser un lenguaje integrado eso beneficia mucho en compiladores de código como Web Assembler entre muchos más que benefician esta investigación. En los últimos años el lenguaje se ha hecho muy popular, gracias a varias razones como:

La cantidad de librerías que contiene, tipos de datos y funciones incorporadas en el propio lenguaje, que ayudan a realizar muchas tareas habituales sin necesidad de tener que programarlas desde cero.

La sencillez y velocidad con la que se crean los programas. Un programa en Python puede tener de 3 a 5 líneas de código menos que su equivalente en Java o C.

La cantidad de plataformas en las que se puede desarrollar, como Unix, Windows, OS/2, Mac, Amiga y otros.

Además, Python es gratuito, incluso para propósitos empresariales, el desarrollo de Python duró varios años, durante los cuales trabajaron diversas compañías de Estados Unidos. En el 2000 ya disponía de un producto bastante completo y un equipo de desarrollo con el que se había asociado incluso en proyectos empresariales. Actualmente trabaja en Zope, una plataforma de gestión de contenidos y servidor de aplicaciones para el web, por supuesto, programada por completo en Python. (Alvarez, 2015) Python está en movimiento y en pleno desarrollo, pero ya es una realidad y una interesante opción para realizar todo tipo de programas que se ejecuten en cualquier máquina en este proyecto se lo implementará al momento de ejecutar las librerías para el sistema de reconocimiento facial.

## **2.3 Marco Conceptual.**

### **2.3.1 AdaBoost.**

AdaBoost entrena de forma secuencial un conjunto de aprendices débiles a partir de un algoritmo base común. Todos los aprendices son entrenados con el mismo conjunto de datos, pero éstos van recibiendo pesos que dependen de los errores cometidos por cada aprendiz. AdaBoost (Adaptive Boosting) fue propuesto por Freund y Schapire en 1995, y consiste en crear varios predictores sencillos en secuencia, de tal manera que el segundo se ajusta bien

lo que el primero no ajustó, que el tercero ajuste un poco mejor lo que el segundo no pudo ajustar y así sucesivamente (Therneau, 2019).

El manejo de AdaBoost es el algoritmo sencillo base en el reconocimiento de patrones que es el fundamental en el algoritmo de reconocimiento que utiliza Opencv.js para detectar objetos, rostros, etc. En la siguiente figura 3 se muestra una ilustración de lo que es AdaBoost.

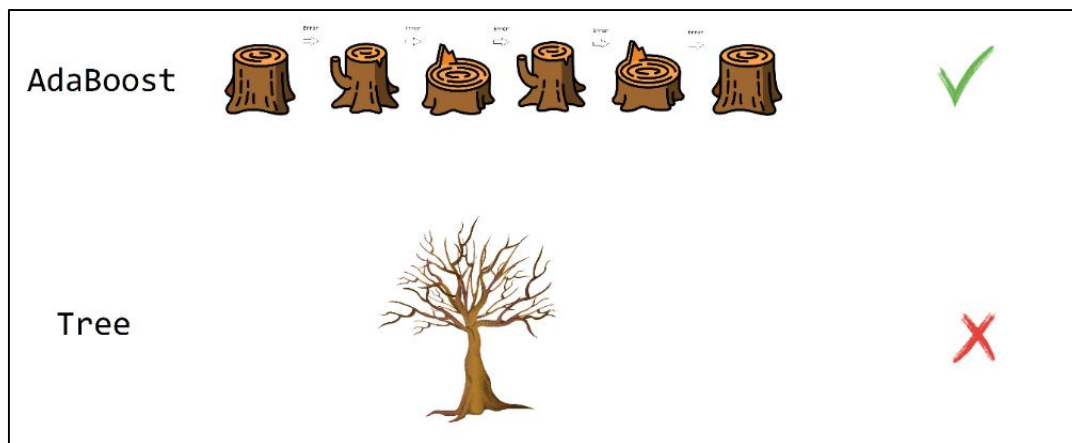


Figura 6 Representación Adaboost. Información tomada de fhermanb.github.io. Elaborado por el autor.

### 2.3.2 Api.

“Una API es la interfaz que permite el intercambio de información entre dos componentes de software independientes. Una API actúa como intermediaria entre las funciones internas y las externas del software, lo que crea un intercambio de información tan sencillo que a menudo pasa desapercibido ante el usuario final.” (Slate, 2019).

Las API como se conoce comúnmente en la actualidad sirven para usos exclusivos de aportes en empresas grandes para la elaboración de software a medida en menos tiempos como beneficio únicamente de ser la interfaz que comunica entre sistemas.

### 2.3.3 Api REST.

REST cambió por completo la ingeniería de software a partir del 2000. Este nuevo enfoque de desarrollo de proyectos y servicios web fue definido por Roy Fielding, el padre de la especificación HTTP y uno de los referentes internacionales en todo lo relacionado con la Arquitectura de Redes, en su disertación ‘Architectural Styles and the Design of Network-based Software Architectures’. En el campo de las APIs, REST (Representational State

Transfer- Transferencia de Estado Representacional) es, a día de hoy, el alfa y omega del desarrollo de servicios de aplicaciones.

“En la actualidad no existe proyecto o aplicación que no disponga de una API REST para la creación de servicios profesionales a partir de ese software. Twitter, YouTube, los sistemas de identificación con Facebook... hay cientos de empresas que generan negocio gracias a REST y las APIs REST. Sin ellas, todo el crecimiento en horizontal sería prácticamente imposible. Esto es así porque REST es el estándar más lógico, eficiente y habitual en la creación de APIs para servicios de Internet” (BBVAOPEN4U, 2016).

En el presente la implementación de un API REST en sistemas es fundamental para ahorrar tiempos en proyectos que requieran un servicio sin desarrollar desde cero, con el fin de este tipo de servicio se ubican muchas empresas que se dedican a desarrollar servicios que pueden consumir a partir de una cantidad módica de dinero como lo son Microsoft, Google, Amazon, etc.

#### **2.3.4 NodeJs.**

Node.js es un ambiente tecnológico en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como, por ejemplo, servidores web. Fue creado por Ryan Dahl en 2009 y su evolución está apadrinada por la empresa Joyent, que además tiene contratado a Dahl en plantilla.

#### **2.3.5 OpenCv en JavaScript.**

“Web es la plataforma informática abierta más ubicua. Con los estándares HTML5 implementados en todos los navegadores, las aplicaciones web pueden representar videos en línea con etiquetas de video HTML5, capturar videos de cámaras web a través de la API WebRTC y acceder a cada píxel de un cuadro de video a través de la API de lienzo” (OpenCv, 2016). Con la abundancia de contenido multimedia disponible, los desarrolladores web necesitan una amplia gama de algoritmos de procesamiento de imágenes y visión en JavaScript para crear aplicaciones innovadoras". Este requisito es aún más esencial para las aplicaciones emergentes en la web, como la realidad virtual web (WebVR) y la realidad

aumentada (WebAR). Todos estos casos de uso exigen implementaciones eficientes de kernels de visión intensivos en computación en la web.

OpenCV.js es un enlace de JavaScript para un subconjunto seleccionado de funciones OpenCV para la plataforma web. Permite que las aplicaciones web emergentes con procesamiento multimedia se beneficien de la amplia variedad de funciones de procesamiento de imágenes disponibles en OpenCV. OpenCV.js aprovecha Emscripten para compilar funciones OpenCV en asm.js o destinos WebAssembly, y proporciona API de JavaScript para que la aplicación web acceda a ellas. Las futuras versiones de la biblioteca aprovecharán las API de aceleración que están disponibles en la Web, como SIMD y ejecución multiproceso.

### **2.3.6 Web Assembly.**

WebAssembly es un nuevo tipo de código que puede ser ejecutado en navegadores modernos — es un lenguaje de bajo nivel, similar al lenguaje ensamblador, con un formato binario compacto que se ejecuta con rendimiento casi nativo y provee un objetivo de compilación para lenguajes como C/C++ y Rust que les permite correr en la web. También está diseñado para correr a la par de JavaScript, permitiendo que ambos trabajen juntos.

### **2.3.7 Emscripten.**

“Emscripten es una cadena de herramientas para compilar en asm.js y WebAssembly, construida con LLVM, que le permite ejecutar C y C++ en la web a una velocidad casi nativa sin complementos” (Emscripten, 2015).

Emscripten es una herramienta muy utilizada por OpenCv para el manejo de datos a nivel de ensamblador porque incluye motores de velocidad de reconocimiento casi iguales a sistemas creados a partir de C y C++ que usan también esta librería por defecto.

### **2.3.8 Base de Datos.**

“MySQL basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento” (Robledano, 2019) . El uso de la base de datos para el registro de personas en el sistema es una parte fundamental en el proceso de reconocimiento de rostros, usar una base de datos robusta como lo es MYSQL.

## 2.4 Definiciones Conceptuales.

**PowerShell:** Herramienta de comandos creado por Windows para el manejo de comandos del sistema.

**MySQL:** Es un sistema de gestión de base de datos relacional.

**Reconocimiento Facial:** Es un sistema computacional que permite detectar o reconocer los rostros de las personas de manera automática en tiempo real.

**Procesamiento de imágenes:** Es un conjunto de procesos o técnicas que permite descubrir o interpretar las imágenes por medio de una herramienta de visión computacional.

**Inteligencia Artificial:** La Inteligencia artificial es el campo científico de la informática que se centra en la creación de programas y mecanismos que pueden mostrar comportamientos considerados inteligentes

**WebVR:** Es una especificación abierta que permite experimentar la realidad virtual en su navegador. El objetivo es hacer que sea más fácil para todos tener experiencias de realidad virtual, sin importar el dispositivo que tenga.

**HTML5:** Lenguaje de etiquetas, es la base de los sistemas web y derivados.

**JavaScript:** Lenguaje de programación ligero, interpretado y compilado justo a tiempo al de una consola.

**GET:** Es un método que envía información por url estando limitada a 2000 caracteres de información.

**POST:** Es un método que envía información por url, en este caso no tiene límites de caracteres a comparación con GET, y su beneficio es la encriptación de datos de la trama.

**REQUEST:** Permite acceso a toda la información que pasa desde el navegador del cliente al servidor.

**RESPONSE:** Es el objeto más utilizado de todos, ya que sirve para presentar en la pantalla del navegador del cliente el resultado de cualquier código.

**RUST:** Es un lenguaje de programación compilado de propósito general y multiparadigma que sigue en desarrollo por Mozilla.

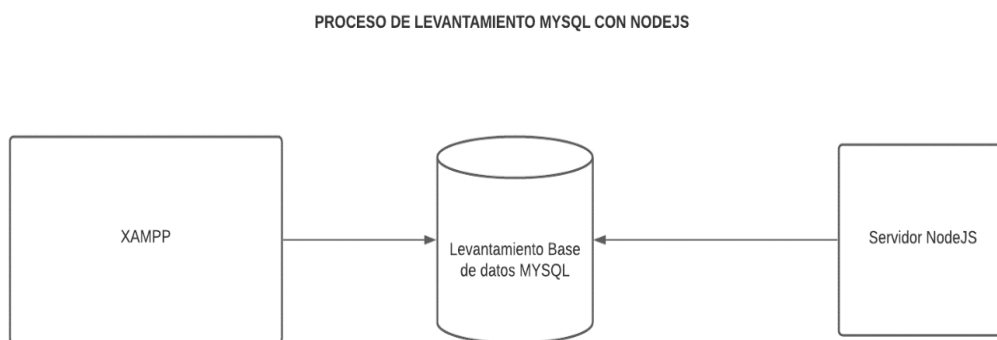
**SIMID:** Interfaz de videos seguros interactivos en plataformas web.

**KERNEL:** Es la parte central o el corazón de cualquier sistema operativo.

## 2.5 Marco Contextual.

### 2.5.1 Demostración de desarrollo API.

Este trabajo de investigación está enfocado en el control de asistencia a estudiantes, hoy en día no existen sistemas que usen un API de reconocimiento facial en ambientes educativos especialmente ahora que la asistencia se controla mediante plataformas web educativas. La base del diseño de este proyecto es el servidor, el levantamiento del proceso es sencillo, con comandos en consola en PowerShell usando la guía que da la documentación de NODEJS en el anexo 1. El proceso de levantamiento del server consta del siguiente como se observa en la figura 6.



*Figura 7 Proceso de levantamiento de server MySQL con Nodejs Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.*



Uno de los procesos del API de reconocimiento facial es la previa subida de información de las personas que van a ser reconocidas mediante un formulario web hecho en NODEJS para el reconocimiento del rostro como se ve en la figura 8.

Figura 8 Formulario Ingreso de información de usuarios. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

La subida de archivos es mediante el explorador de Windows en archivos previamente guardados en el ordenador este proceso se ve en la figura 9.

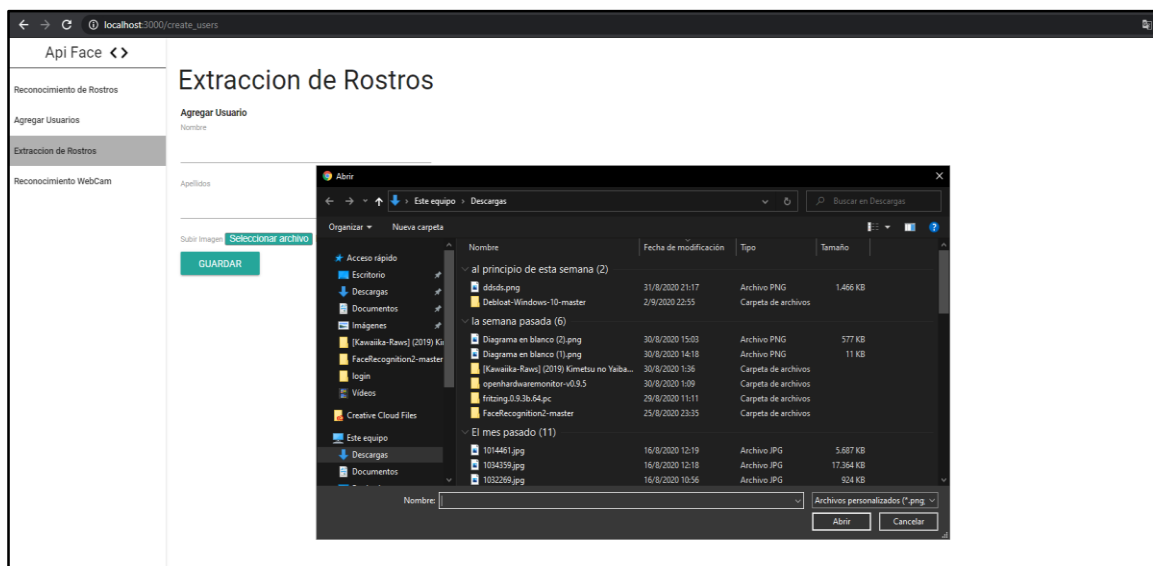


Figura 9 Subida de Datos de personas mediante el explorador de Windows. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

El guardado de imágenes mediante el formulario de subida con el atributo “ENCTYPE” que permite encriptar las imágenes mediante el método de subida POST.

Previo al guardado de la imagen en el segundo proceso la extracción de cada rostro, una vez registrado usando una imagen referencial de una clase en Microsoft Teams en donde se reconocen todos los rostros de la persona como lo en la figura 10.

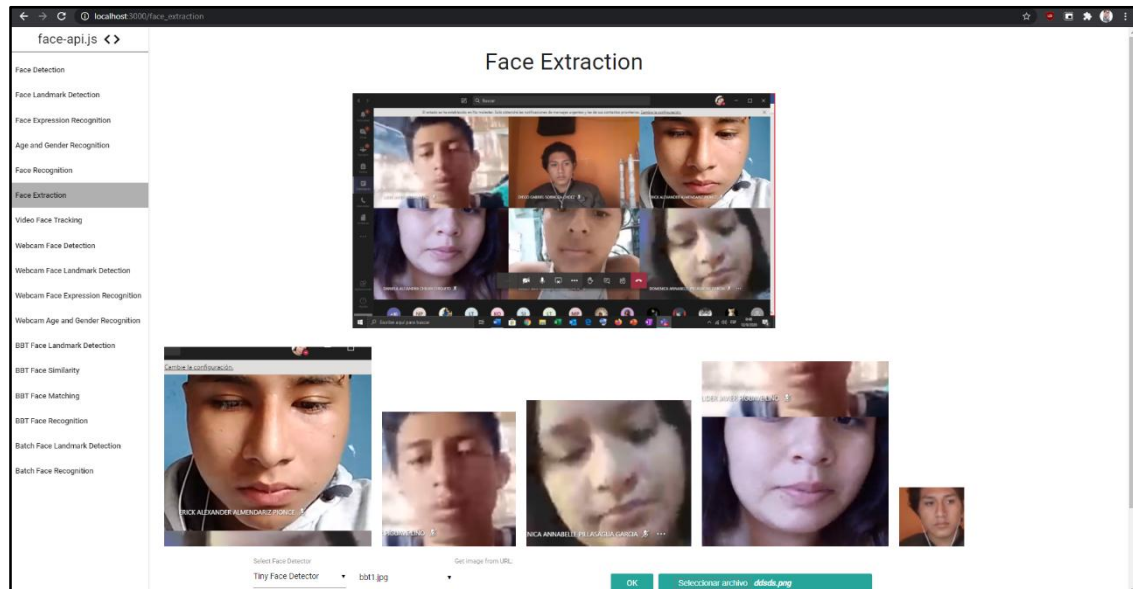


Figura 10 Extracción de rostros por NodeJs. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

Luego de la extracción de los rostros mediante una ruta en GET con el url “localhost/public/obtener” obtener los datos de cada persona reconocida en formato JSON mediante el uso del programa “POSTMAN” como a continuación en la figura 11.

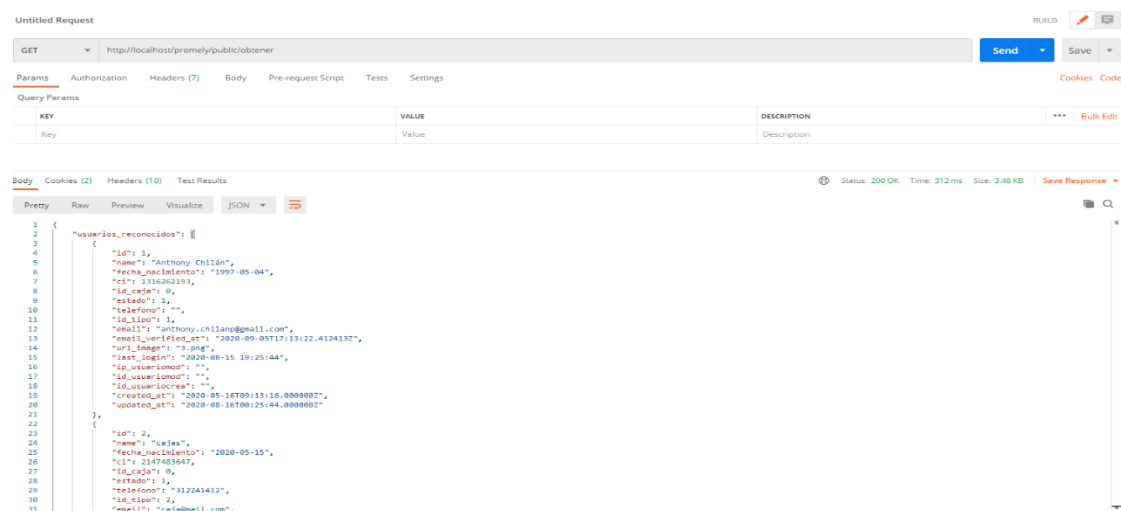


Figura 11 Repuesta a usuarios reconocidos. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

La respuesta en JSON es la mejor opción de método de transmisión para las API de reconocimiento facial por su maquetado en la información, mejor consulta y mejor manejo de datos de respuesta al servidor. Existen diferentes formas de usar este formato como consumirlo en un server o medios parecidos, para búsquedas de datos, etc.

También se puede usar el URL como manera de verlo en navegador web como en la figura 12 a continuación.



```
{
  "usuarios_reconocidos": [
    {
      "id": 1,
      "name": "Anthony Chil\u00e1n",
      "fecha_nacimiento": "1997-05-04",
      "ci": "1316262193",
      "id_caja": 0,
      "estado": 1,
      "telefono": "",
      "id_tipo": 1,
      "email": "anthony.chilanp@gmail.com",
      "email_verified_at": "2020-09-05T18:06:02.883586Z",
      "url_image": "3.png",
      "last_login": "2020-08-15T19:25:44",
      "ip_usuariomod": "",
      "id_usuariomod": "",
      "id_usuariocrea": "",
      "created_at": "2020-05-16T09:13:16.000000Z",
      "updated_at": "2020-08-16T00:25:44.000000Z"
    },
    {
      "id": 2,
      "name": "cajas",
      "fecha_nacimiento": "2020-05-15",
      "ci": "2147483647",
      "id_caja": 0,
      "estado": 1,
      "telefono": "312241412",
      "id_tipo": 2,
      "email": "caja@mail.com",
      "email_verified_at": "2020-09-05T18:06:02.884439Z",
      "url_image": "3.png",
      "last_login": "2020-07-05T21:56:25",
      "ip_usuariomod": "",
      "id_usuariomod": "",
      "id_usuariocrea": "",
      "created_at": "2020-05-16T21:39:33.000000Z",
      "updated_at": "2020-07-06T02:56:25.000000Z"
    },
    {
      "id": 3,
      "name": "liam",
      "fecha_nacimiento": null,
      "ci": null,
      "id_caja": null,
      "estado": 1,
      "telefono": null,
      "id_tipo": 2,
      "email": "anthony.chilanps@gmail.com",
      "email_verified_at": null,
      "url_image": null,
      "last_login": null,
      "ip_usuariomod": null,
      "id_usuariomod": null,
      "id_usuariocrea": null,
      "created_at": "2020-07-30T02:38:16.000000Z",
      "updated_at": "2020-07-30T02:38:16.000000Z"
    },
    {
      "id": 4,
      "name": "liam2",
      "fecha_nacimiento": null,
      "ci": null,
      "id_caja": null,
      "estado": 1,
      "telefono": null,
      "id_tipo": 2,
      "email": "anthony.chilan124@gmail.com",
      "email_verified_at": null,
      "url_image": null,
      "last_login": null,
      "ip_usuariomod": null,
      "id_usuariomod": null,
      "id_usuariocrea": null,
      "created_at": "2020-07-30T03:01:33.000000Z",
      "updated_at": "2020-07-30T03:01:33.000000Z"
    },
    {
      "id": 5,
      "name": "liam2",
      "fecha_nacimiento": null,
      "ci": null,
      "id_caja": null,
      "estado": 1,
      "telefono": null,
      "id_tipo": 2,
      "email": "prueba@mail.com",
      "email_verified_at": null,
      "url_image": null,
      "last_login": null,
      "ip_usuariomod": null,
      "id_usuariomod": null,
      "id_usuariocrea": null,
      "created_at": "2020-07-30T03:02:55.000000Z",
      "updated_at": "2020-07-30T03:02:55.000000Z"
    },
    {
      "id": 6,
      "name": "liam",
      "fecha_nacimiento": null,
      "ci": null,
      "id_caja": null,
      "estado": null,
      "telefono": null,
      "id_tipo": 2,
      "email": "cajas@mail.com",
      "email_verified_at": null,
      "url_image": null,
      "last_login": null,
      "ip_usuariomod": null,
      "id_usuariomod": null,
      "id_usuariocrea": null,
      "created_at": "2020-07-30T03:17:40.000000Z",
      "updated_at": "2020-07-30T03:17:40.000000Z"
    },
    {
      "id": 7,
      "name": "kennie",
      "fecha_nacimiento": null,
      "ci": null,
      "id_caja": null,
      "estado": null,
      "telefono": null,
      "id_tipo": null,
      "email": "prub@mail.com",
      "email_verified_at": null,
      "url_image": null,
      "last_login": null,
      "ip_usuariomod": null,
      "id_usuariomod": null,
      "id_usuariocrea": null,
      "created_at": null,
      "updated_at": null
    }
  ]
}
```

*Figura 12 Respuesta personas mediante el navegador. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.*

## 2.6 Marco Legal

### 2.6.1 Artículos Investigativos

Este proceso se detallará un breve resumen basado en los artículos investigativos de la Ley Orgánica Europea y Ley Intelectual, Uso del Software Libre que se encontrarán en los anexos que son muy importante para el desarrollo de este trabajo de titulación.

**Tabla 2** Artículos Investigativos.

Artículos Investigativos.
En el anexo 2, se detallarán los artículos basado al reconocimiento facial para identificar el rostro facial del estudiante detectándolo en tiempo real, por medio de visión artificial, dicho artículo 9 y 14 de la “Ley Orgánica Europea” indica sobre la protección de los datos por medio de biométricos, Vision Computacional deben ser muy clara por la ley. López explica que sin perjuicio de que el usuario debe ser correctamente informado sobre el

---

tratamiento de datos personales, incluido los adquiridos de rasgos faciales. Fuente especificada no válida.

Sánchez explica que el artículo 28 de la “Ley Intelectual” indica que los programas requieren de una protección de datos que deben ser protegidos para la adquisición del ordenador. Fuente especificada no válida.

---

*Información tomada de investigación previa de la Ley Orgánica de la Unión Europea*

**Tabla 3** Uso de Software Libre en el Ecuador

<b>Uso de Software Libre en el Ecuador</b>
De no indica que aquellos sistemas y equipamientos informáticos dado por el Decreto 104 del Gobierno Ecuatoriano se debe permitir el acceso de código y aplicaciones que deben ser mejoradas, cada software libre es propietario cuando se satisfagan dichos requerimientos, para así poder verificar la existencia y la capacidad técnica que brinde el soporte de dicho sistema, dichos artículos se detallarán en el anexo 3. <b>Fuente especificada no válida..</b>
<i>Información tomada de investigación previa del Reglamento Interno del Decreto Ejecutivo N. 104.</i>

#### **Artículos que existen dentro del Reglamento Jurídico de la República del Ecuador.**

Estos artículos están basados en la educación estudiantil, a los jóvenes, docentes y usos de programas a utilizar en el desarrollo de tesis, se debe tener en cuenta que cada ciudadano tanto los docentes y los jóvenes deben ser capacitados mediante el desarrollo de nuevas tecnologías, que puedan demostrar enseñanzas adquiridas dentro del proceso de formación académica y profesional que debe cumplir todo ser humano. Este proyecto va de acuerdo a los reglamentos Jurídico de la República del Ecuador como son: Constitución de la República del Ecuador, Buen Vivir, Educación Superior y Código Orgánico Penal.

**Tabla 4** Constitución de la República del Ecuador

<b>Constitución de la República del Ecuador</b>
En el anexo 4, se detallarán los artículos de la Constitución de República del Ecuador, nos indica que cada persona o familia tendrán el derecho y la responsabilidad de participar en el proceso educativo para prepararnos profesionalmente y de excelencia académica

dentro de la educación. En los jóvenes tiene derecho de prepararse para ser mejores profesionales dentro del desarrollo del país. **Fuente especificada no válida.**

*Información tomada de Investigación previa Constitución de la República del Ecuador.*

**Tabla 5** Régimen del Buen Vivir

<b>Régimen del Buen Vivir</b>
En el anexo 5, se describen los artículos de Buen Vivir dentro de campo de Ciencia, tecnológica y saberes ancestrales que se debe generar y difundir los conocimientos adquiridos en el aprendizaje tanto científicos y tecnológicos, dichos sistemas comprenderán programas que deben ser revisados por las autoridades que dentro del desarrollo de la innovación para el bienestar de los ciudadanos. <b>Fuente especificada no válida..</b>
<i>Información tomada de Investigación previa del Régimen del Buen Vivir.</i>

**Tabla 6** Ley Orgánica de Educación Superior

<b>Ley Orgánica de Educación Superior</b>
En el anexo 6, se detallarán los artículos basado en la Educación Superior que cada estudiante tiene el derecho de estudiar para desempeñar profesionalmente por el futuro del país y lograr méritos académicos. Los profesores de cada institución tienen el derecho de investigar, garantizar la estabilidad promocional, movilidad basado en méritos académicos, la calidad de enseñanzas sin afectar a las demás instituciones. <b>Fuente especificada no válida..</b>
<i>Información tomada de Investigación previa de la Ley Orgánica de la Educación Superior</i>

**Tabla 7** Código Orgánico Penal

<b>Código Orgánico Penal</b>
En el anexo 7 se mostrarán los artículos basado las precauciones del acceso permitido dentro del sistema informático, telemático o de telecomunicaciones, cada persona que acceda sin autorización alguna como modificar un portal web, desviar tráfico de datos, entre otros será sancionado por 5 años. (COIP, 2014).
<i>Información tomada de Investigación previa del Código Orgánico Penal.</i>

## Capítulo III

### Metodología

#### 3.1. Modalidad de la Investigación

Con la finalidad de tramitar la problemática planteada anteriormente se propone el diseño del API para detectar los rostros de los estudiantes de la facultad de Tecnología de Información de la ciudad de Jipijapa esta herramienta mostrara la eficiencia en sistemas de reconocimiento facial al detectar rostros y tomar la asistencia en el cual este va ser una herramienta en beneficio a los profesores cuyo sistema facilitara el proceso normal en la toma de asistencias.

**Tabla 81** Variable de la Investigación

Variable de la investigación			
Tipo de variable	Variable	Dimensiones	Indicadores
Independiente	API de reconocimiento facial para control de asistencia a estudiantes	A nivel de educación superior en la facultad de tecnología de información	Proceso de mejora en la toma de asistencias.
Dependiente	Toma de asistencia a estudiantes en la facultad de Tecnología de información	A nivel de la Carrera de licenciatura	Proceso de mejora en la toma de asistencias.

*Información tomada de la investigación directa. Elaborado por Anthony Limber Chilán Pincay.*

#### 3.2. Enfoque de la investigación

Para el diseño del API, se considera los siguientes aspectos importantes que son:

- Investigación Técnica
- Investigación Cuantitativa
- Investigación Cualitativa

Debido al estudio de esta investigación de los diferentes sistemas de control de asistencia, existen distintos tipos de reconocimiento facial que ayudan a sistemas de control de accesos

y actualmente no existen sistemas que implementen en comunicación con sistemas actuales educativos que se tiene.

### 3.2.1. Investigación Técnica

En base a las pruebas de diseño se realizó una simulación durante el proceso de titulación en la que se determinó lo viable del uso de herramientas open source (código abierto), estas herramientas tecnológicas brindan una mejor viabilidad al momento de desarrollar dicho API con las funcionalidades que requieren para el control de asistencia de estudiantes. Los recursos para la funcionalidad tanto como en hardware o software se encuentran en la mayor parte del mercado actual.

El uso de estas herramientas comprende de dos etapas una de ellas el backend que básicamente compone el cuerpo del API, y el frontend como lo es NODEJS. Uno de los componentes del backend es MySQL en la cual se almacena los datos de los estudiantes para su detección. MySQL es un gestor que viene incluido con XAMP, mediante esto facilita el manejo a nivel de servidor y gestión de datos de la base de datos del API.

En el frontend se usará NODEJS que es un intermedio de comunicación entre la parte del backend y el frontend, que usara los datos que guarda y los mostrara en una página basada en HTML y JavaScript, y por último para la parte principal del proyecto para el reconocimiento de los rostros en donde se usa OpenCv con JavaScript.

**Tabla 9** Recurso de Software

Herramientas	Característica
SO (Sistema Operativo)	Multisistema portable
Base de Datos	MySQL
Servidor de Aplicaciones	Apache/ NodeJs
Framework	Bootstrap, JQuery
Lenguaje de Programación	TypeScript/ JavaScript

*Información tomada de la investigación directa. Elaborado por Anthony Limber Chilán Pincay*

Una de las ventajas de usar este tipo de tecnologías de código abierto es lo escalable que puede ser usar Nodejs y los beneficios de su escalabilidad a nivel estructural facilitan como

anteriormente en la tabla 9 sobre el recurso del software. El gestor de base de datos amigable adaptable a múltiples IDE para el manejo de información.

Servidor web de Nodejs que da como ventaja ser asíncrono en comunicación con el backend y el frontend. La experiencia UIX a nivel de usuario que brinda Bootstrap.

En la tabla 10 se detalla uno de los beneficios a nivel de hardware que brinda Nodejs como servidor.

**Tabla 10** Recurso Hardware – Servidor

Características	Especificaciones
Procesador	Core i3 6ma Generación
Memoria	4 Gb RAM
Disco Duro	1 TB
Monitor "15"	Portátil
Periféricos	Teclado, Mouse, Cámara Web

*Información tomada de la investigación directa. Elaborado por Anthony Limber Chilan Pincay*

### 3.2.2. Investigación Cuantitativa

El uso de esta información se obtiene la investigación directa, que con el uso de esta metodología permitirá recopilar, analizar la información por medio de datos de un estudio previamente hecho, las herramientas tecnológicas que se usan en este diseño ayudaran a obtener resultado sobre el funcionamiento de la simulación del API.

### 3.2.3. Investigación Cualitativa

El docente usara la aplicación que disponen el conocimiento necesario para el funcionamiento del sistema, se realizara una encuesta a los docentes con dichas preguntas elaboradas de manera objetiva con la finalidad de enfocarse en el diseño del API.

## 3.3. Población

La población es el conjunto de personas que se encuentran en un sitio determinado, me permitirá realizar generalizaciones debido al tamaño de Docentes de la Carrera de Tecnología de Información de la Universidad Estatal del Sur de Manabí de la ciudad de



Jipijapa mediante el uso del muestreo no probabilístico, se va calcular de la siguiente manera:

$$n = \frac{N}{1 + \frac{e^2(N-1)}{z^2pq}}$$

**Dónde:**

**n**= tamaño de la muestra

**N**= tamaño conocido de la población

**z**= nivel de confianza 92%

**e**= error máximo permitido 6%

**p**= Probabilidad a favor

**q**= Probabilidad en contra

**Datos:**    **N=180**    **z= 1.96**    **e= 6**    **p= 0.50**    **q= 0.50**    **n=?**

$$n = \frac{180}{1 + \frac{0.05^2(180-1)}{1.96^2(0.50)(0.50)}}$$

$$n = \frac{180}{1 + \frac{0.0025(179)}{3.8416(0.25)}}$$

$$n = \frac{180}{1 + \frac{0.0475}{0.9604}}$$

$$n = \frac{180}{1.0494}$$

$$n = 98$$

Por lo tanto, será necesario encuestar no menos de 98 usuarios para tener una fiabilidad en la información del 92%.

Durante el procesamiento de la información la cantidad de información que se obtuvo en la encuesta propuesta, en todas las descripciones se hicieron con la herramienta de formulario de Google que da beneficios como estadísticas de la encuesta en porcentaje de respuestas, representaciones en tablas y graficas que se necesitan para el análisis de interrogantes.

### 3.3.1. Ejecución de encuestas

Aplicando el método de investigación cuantitativa, se procedió a utilizar el instrumento encuesta, mediante la cual la misma fue enviada a los docentes explicando previamente los conceptos de API y el objetivo del diseño.

El desarrollo de la encuesta como se mencionó con anterioridad se estableció como población a docentes de una institución educativa en la cual se realizó el proceso de recolección de información.

### 3.3.2. Análisis de las encuestas

Este modelo va dirigido a docentes en general que usan el control de asistencia de manera manual o usando apartados de sistemas educativos en general. Este modelo permitirá conocer una posible demanda en beneficio a la prueba y simulación de este diseño dentro de la ciudad de Jipijapa.

#### Resultados de la encuesta

Se realizó el proceso de encuesta a 20 Docentes que mostraron su punto de vista de acuerdo al beneficio del diseño de este proyecto con finalidad de obtener resultados para llegar a las conclusiones debidas. Con el fin a este proceso se realizó 8 preguntas como se indica en el anexo 8, en consecuencia, a esto las preguntas son las siguientes:

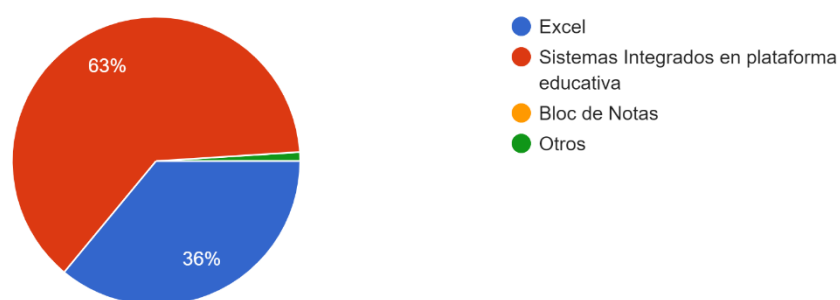
#### 1. ¿Qué método usa usted frecuentemente para la toma de asistencia a estudiantes?

**Tabla 21** Métodos de toma de asistencia

Opciones de Respuestas	Docentes	Porcentajes
Excel	63	63%
Sistemas Integrados en plataforma educativa	36	36%
Bloc de Notas	0	0%
Otros	1	1%
<b>Total</b>	<b>100</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas



*Figura 13 Métodos de toma de asistencia a estudiantes, Información tomada de la investigación directa. Elaborado por Anthony Limber Chilán Pincay*

Según los datos que se obtuvieron en la figura 13 de la pregunta 1, se dio como resultado que los docentes usan en su mayoría (65%) como método de toma de asistencia la plataforma educativa o Excel, solo el 35% emplea otros métodos de tipo manual o semi manual.

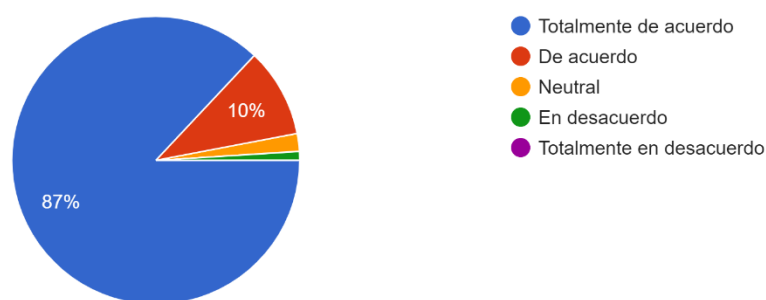
## 2. ¿Considera usted eficiente el proceso de detección de rostros para el control de datos de una persona?

**Tabla 32** Proceso de detección de rostros

Opciones de Respuestas	Docentes	Porcentajes
Totalmente de Acuerdo	81	81%
De Acuerdo	17	17%
Neutral	4	4%
En desacuerdo	0	0%
Totalmente en desacuerdo	2	2%
<b>Total</b>	<b>20</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas



*Figura 14 Proceso de detección de rostros. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.*

En los resultados obtenidos en la figura 15 de la pregunta 2, la mayoría de los docentes (80%) está de acuerdo con el proceso de detección de rostros, solo el 20% le es indiferente o no está de acuerdo.

### 3. ¿Tiene usted conocimiento sobre visión artificial para el proceso de detección de rostros?

**Tabla 43** Visión Artificial para proceso de detección de rostros

Opciones de Respuestas	Docentes	Porcentajes
Mucho	11	55%
Poco	6	30%
Nada	3	15%
<b>Total</b>	<b>20</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas

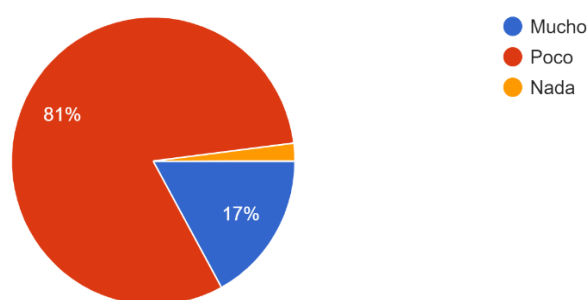


Figura 15 Visual Artificial para proceso de detección de rostros. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

Según la figura 15 de los resultados obtenidos, alrededor de la mitad de los docentes conocen de tecnologías de reconocimiento facial, la otra mitad indica muy poco conocimiento o no conoce sobre este tema.

#### 4. ¿Conoce usted una aplicación que implemente reconocimiento de rostros?

**Tabla 54** Aplicación de reconocimiento de rostros

Opciones de Respuestas	Docentes	Porcentajes
Si	87	87%
No	13	13%
<b>Total</b>	<b>100</b>	<b>100%</b>

Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.

100 respuestas

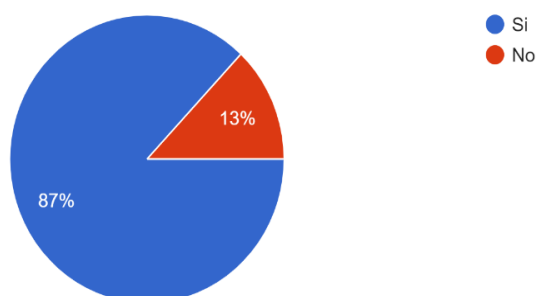


Figura 16 Aplicación de reconocimiento de rostros. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

Un gran porcentaje de los docentes encuestados indicaron que tienen el conocimiento de aplicaciones que implemente este tipo de tecnología. Esto permite percatar de una posibilidad de mejor comprensión en lo que se refiere a este proyecto.

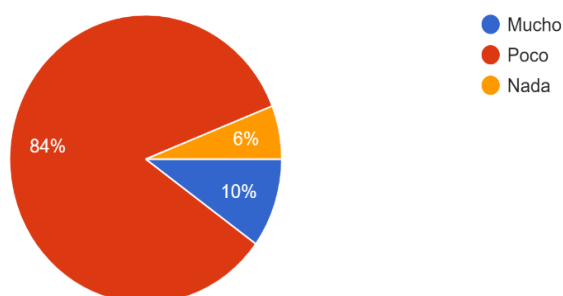
### 5. ¿Tiene usted conocimiento sobre el uso de un API para detección de rostros?

**Tabla 65** Uso API para detección de rostros

Opciones de Respuestas	Docentes	Porcentajes
Mucho	10	10%
Poco	84	84%
Nada	6	6%
<b>Total</b>	<b>100</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas



*Figura 17* Uso API para detección de rostros. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

Un bajo porcentaje de los docentes indican tener desconocimiento sobre las tecnologías denominadas API (10%), el 84% menciono tener muy poco conocimiento o ninguno.

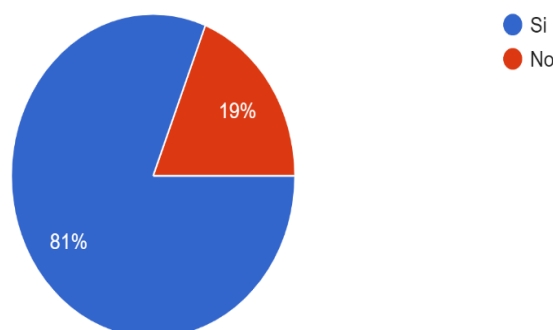
**6. ¿Actualmente los docentes cuentan con una herramienta de reconocimiento de rostros para el control de asistencia a estudiantes?**

**Tabla 76** Herramienta de rostros para el control de asistencia

Opciones de Respuestas	Docentes	Porcentajes
Si	11	52.4%
No	9	47.6%
<b>Total</b>	<b>20</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas



*Figura 18* Herramienta de rostros para el control de asistencia. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

Como se observa en la figura 17, en un 81% de la población docente encuestada indico que, si cuenta con la herramienta, y la otra dijo que no cuentan con la herramienta tecnológica para la toma de asistencia.

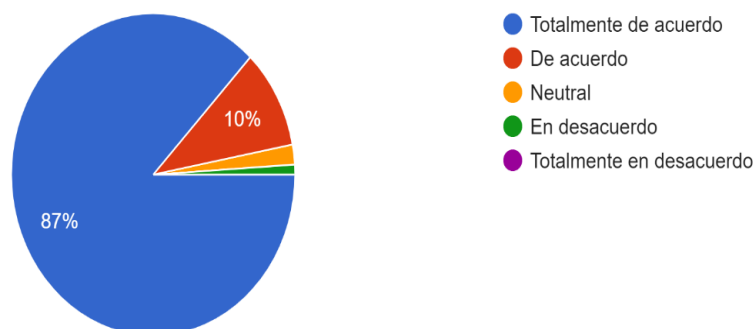
**7. ¿Considera eficiente usar tecnologías de reconocimiento facial para el control de asistencia a estudiantes en aulas virtuales o presenciales?**

**Tabla 87** Tipos de tecnología para el control de asistencia

Opciones de Respuestas	Docentes	Porcentajes
Totalmente de Acuerdo	87	87%
De Acuerdo	10	10%
Neutral	1	1%
En desacuerdo	1	1%
Totalmente en desacuerdo	1	1%
<b>Total</b>	<b>100</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas



*Figura 19* Tipos de tecnología para el control de asistencia *Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.*

Como se observa en la figura 19, El 97 % dijeron estar de acuerdo con el uso de estas tecnologías, en un 3% respondieron que le es indiferente la propuesta.

**8. ¿Estaría dispuesto a usar un API de reconocimiento facial para el uso y control de asistencia a estudiantes en clases virtuales o presenciales?**

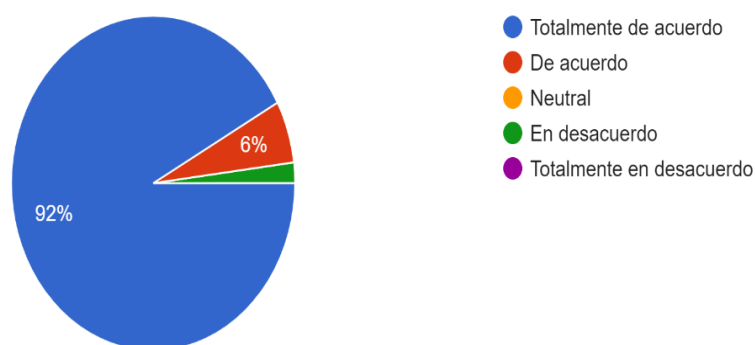


**Tabla 98** Uso API de reconocimiento facial

Opciones de Respuestas	Docentes	Porcentajes
Totalmente de Acuerdo	92	87%
De Acuerdo	6	10%
Neutral	0	0%
En desacuerdo	2	2%
Totalmente en desacuerdo	0	0%
<b>Total</b>	<b>100</b>	<b>100%</b>

*Información tomada de la investigación directa, Elaborado por Anthony Limber Chilán Pincay.*

100 respuestas



*Figura 20* Uso API de reconocimiento facial. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

Según lo encuestado en la figura 20 se puede observar que el 98% de los docentes están de acuerdo a usar el API de reconocimiento facial para el uso de control de asistencia, y un 2% no están de acuerdo de usar este tipo de tecnologías.

### 3.4. Propuesta de investigación

El diseño de este aplicativo implementando reconocimiento facial es uno de los principales puntos en esta investigación lo cual permitirá un manejo de asistencia de estudiantes, además de contar con múltiples beneficios a los usuarios que brinda lenguajes como NodeJs y OpenCv.

El uso de este aplicativo permitirá al docente ahorrar un proceso común en el área educativa además de llevar un control eficiente del mismo.

### 3.5. Esquema general de simulación

Para el desarrollo de una simulación en base al diseño de este API, es necesario ver que se compone de dos partes la parte del backend y la parte del frontend.

El backend se compone de múltiples servicios como se ve en la figura 20



Figura 21 Estructura Backend Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

El frontend que se compone de la parte visual del API como se observa ver en la figura 22.

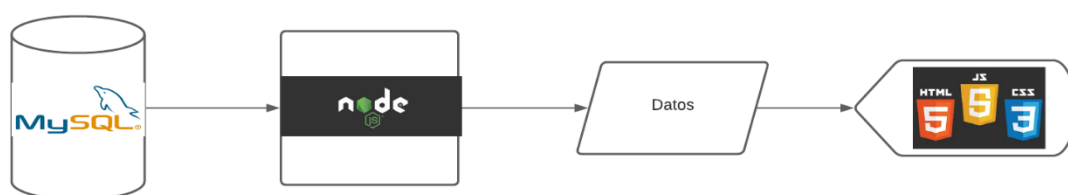


Figura 22 . Estructura Frontend. Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

### 3.6. Servidor XAMPP

Para el backend es necesario el servidor como base de simulación en XAMPP, lo primero sería la instalación como se observa en el anexo 9, partiendo de la instalación se levanta el servidor de MySQL como se observa en la figura 23.

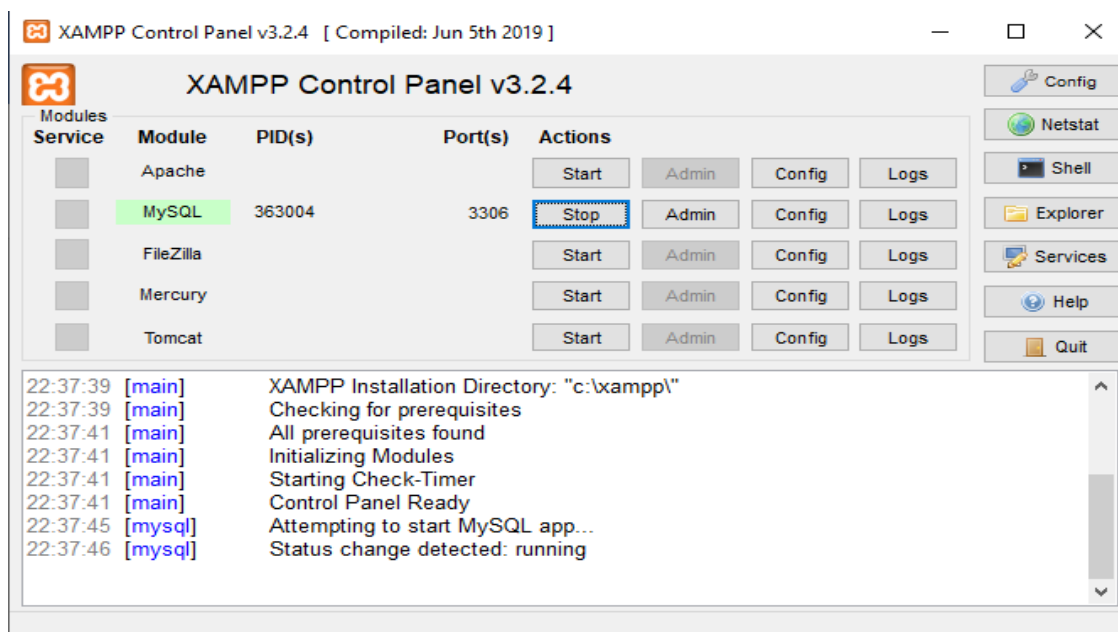


Figura 23 Levantamiento de Servidor MySQL Información tomada de manera directa. Elaborado por Anthony Chilán Pincay

Luego del levantamiento del servidor MySQL se procede a la creación de la base de datos con el nombre “faceapi” como se observa en la figura 24.

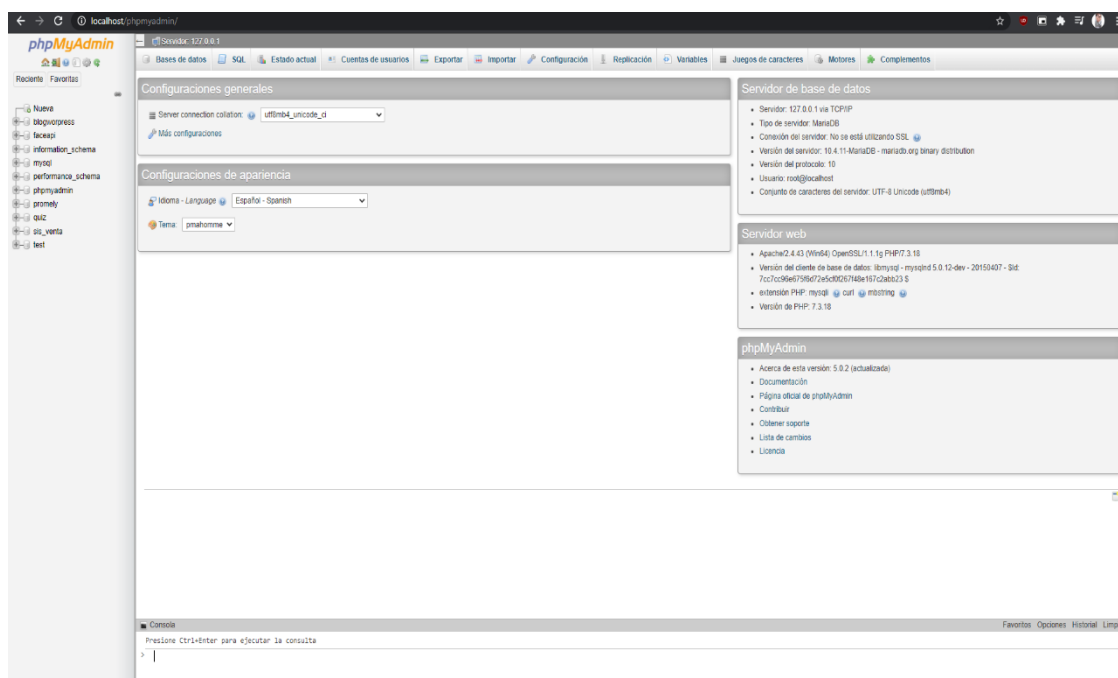


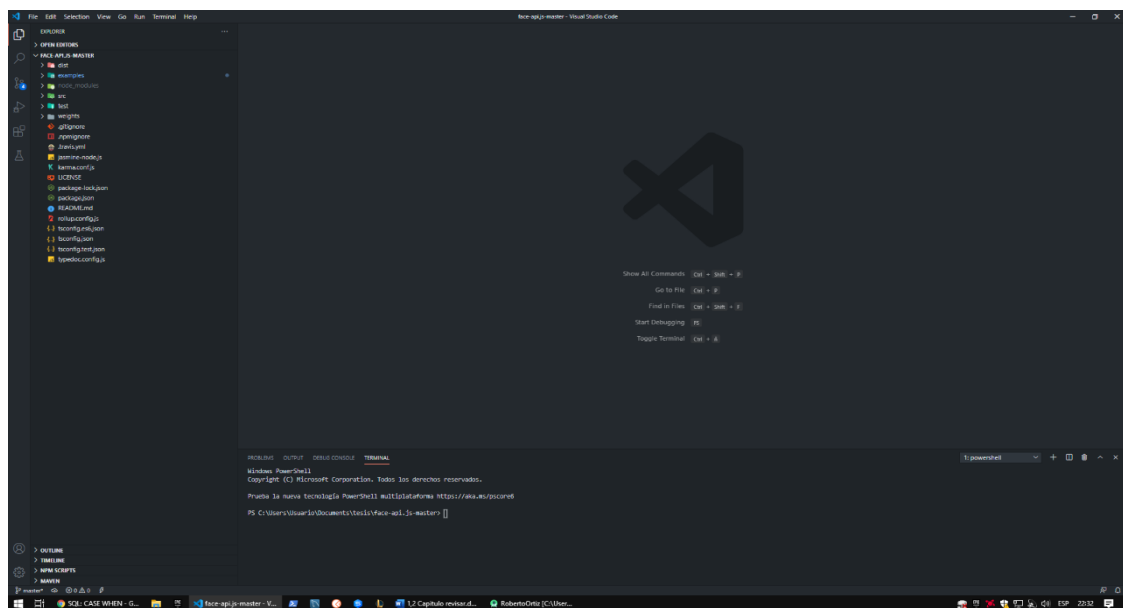
Figura 24 Creación de base de datos. Información tomada directa elaborado por Anthony Chilán Pincay,

### 3.7.Server NodeJs

La comunicación con la base de datos y con el servidor lo compone especialmente NodeJs que tiene el beneficio de conectarse de manera asíncrona con la base de datos para obtener

datos en tiempo real, usando JavaScript como lenguaje base se procede con la conexión con la base de datos como se puede observar en el anexo 10. Además de la conexión con el servidor de MySQL, Nodejs tiene su propia escucha en el sistema para procesos de levantamiento de servidor, para ello es necesario usar el puerto 3000 como se observa en el anexo 11.

El proceso de levantamiento de servidor se puede realizar de 2 formas: Mediante un IDE de desarrollo como Visual Studio Code como se puede observar a continuación en la figura 25.



*Figura 25 IDE de desarrollo Visual Studio Code. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay.*

O Mediante la línea de comando, en este caso como PowerShell como en la figura 26.

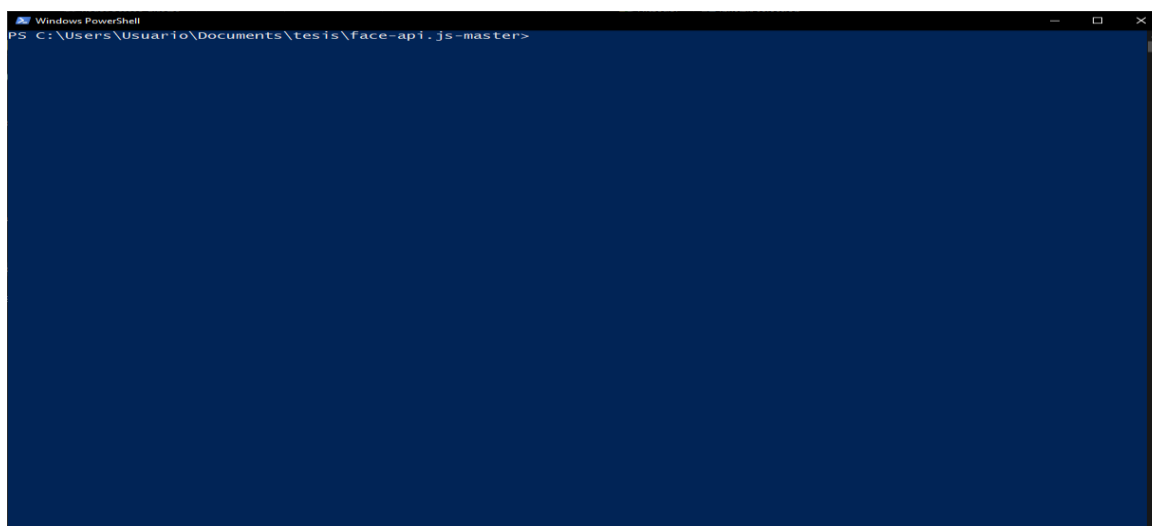


Figura 26 Powershell de Windows. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay.

Para el proceso de creación del servidor en NodeJs basta con simples líneas de código como “npm init” pasando al proceso de creación del proyecto con el nombre de faceapi.js-master mediante estas dos formas se maneja el problema de las dependencias. Una de estas dependencias podría ser la instalación de la librería de MySQL server para el manejo de la base de datos de esta investigación. Para la instalación de esta dependencia se puede observar como el uso del PowerShell o visual code permitirá la rápida instalación con el comando “npm -i” y adjuntado el nombre de la dependencia como en la figura 27.

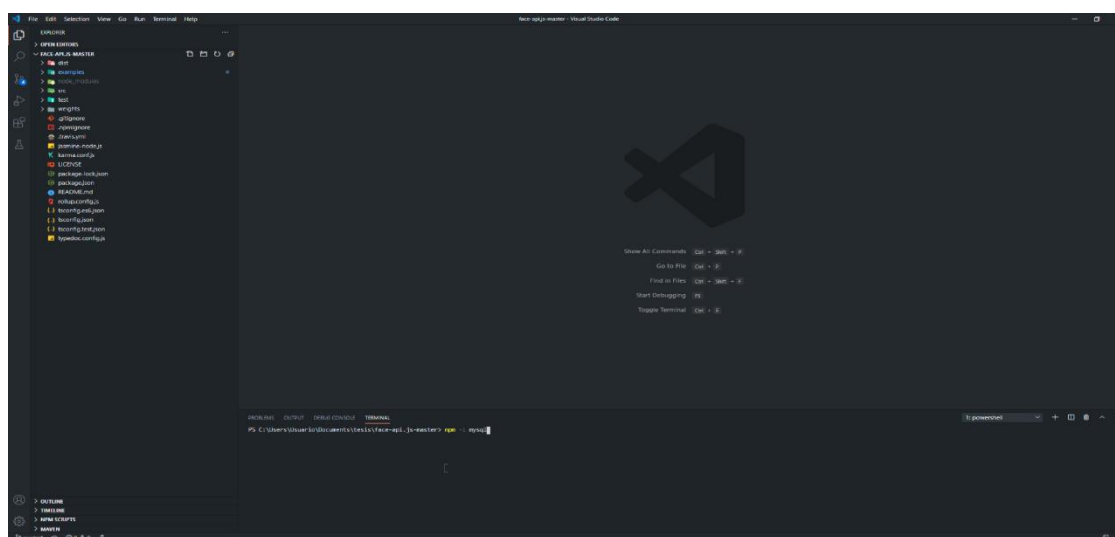


Figura 27 Instalación MySQL en NodeJs Información tomada de la investigación directa. Elaborado por Anthony Chilán Pincay.

### 3.8. Estructura Base de Datos

Para el diseño de esta API la estructura de datos está definida en dos secciones:

La gestión de usuarios propuestas para el control único administrativo, y la gestión del docente como irá el detallado a continuación del modelo en MySQL.

#### 3.8.1. Detalle de la Base de datos

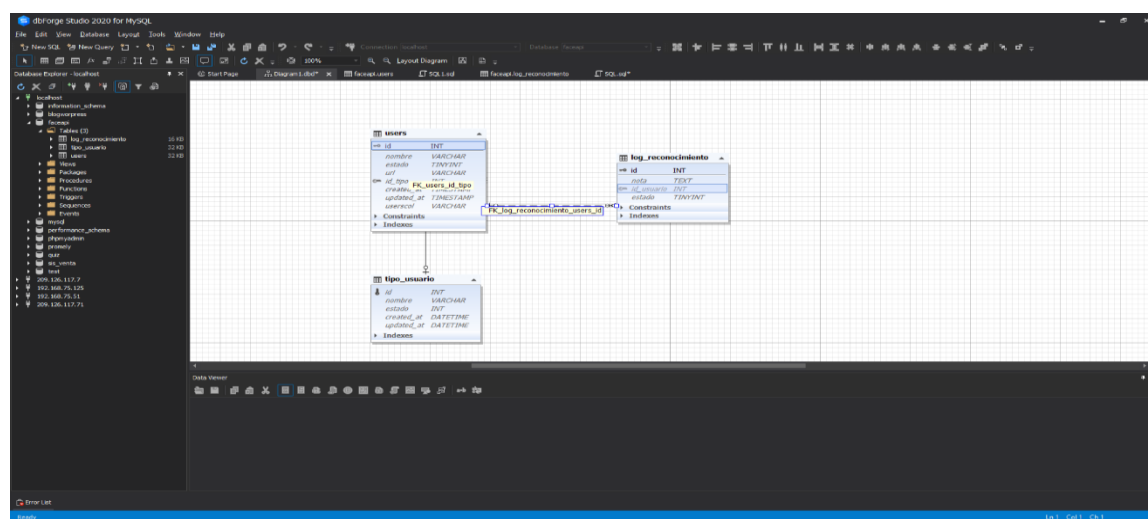
**Tabla 19** Descripción de la Base de datos

Parámetro	Datos
IDE de la Base de datos	dbForge Studio
Dominio	127.0.0.1
Puerto	3400

*Información tomada de la investigación realizada, Elaborado por Anthony Limber Chilán Pincay.*

#### 3.8.2. Modelo de la Base de datos

Según lo mencionado anteriormente al usar la base de datos el modelo lógico de la base de datos se crearía a partir de un IDE y gestor de base de datos llamado “dbForge Studio” el proceso de instalación se puede observar en el anexo 12, y su uso es intuitivo para observar el diagrama de las tablas en la figura 25.



*Figura 28* Modelo de Base de Datos en dbForge Studio. Información tomada directamente de la investigación. Elaborado por Anthony Limber Chilán Pincay.

### 3.8.3. Información de ambiente de Programación

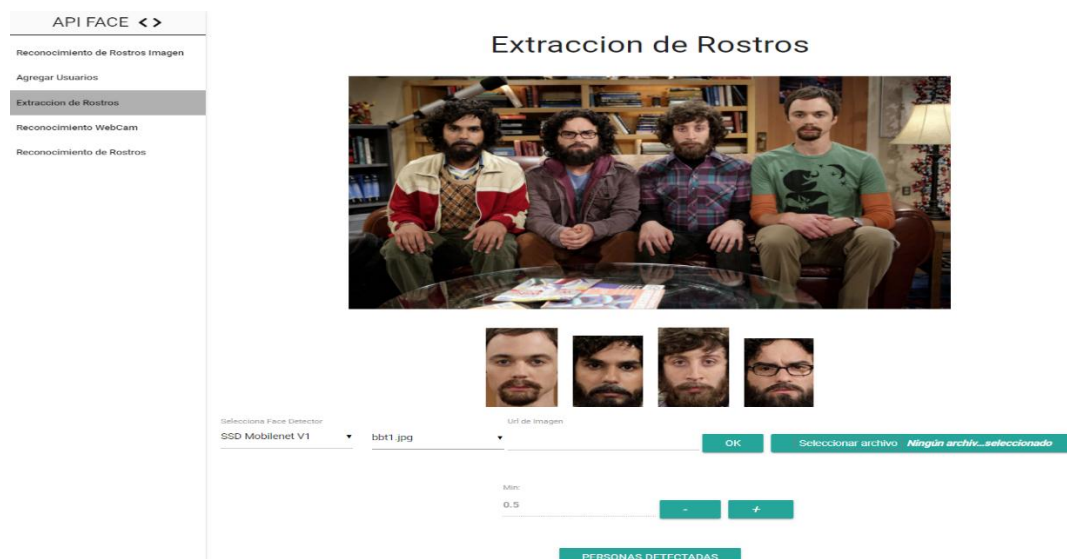
**Tabla 20** Estructura del ambiente de Programación

Parámetro	Datos
Lenguaje de Programación	JavaScript y SQL
Servidor Web	XAMPP y NodeJs
Puerto	8088, 3000
Framework	NodeJs
JavaScript	Integrado

*Información tomada de la investigación realizada, Elaborado por Anthony Limber Chilán Pincay.*

### 3.9. Creación de API de reconocimiento facial

Este API se basa en un diseño web intuitivo con múltiples funcionalidades añadidas en un Sidebar de opciones para el control de asistencia. Al referirse el termino Sidebar se refiere a un menú de opciones múltiples del aplicativo web en el cual se muestran todo tipo de opciones como se observa en la figura 29.



*Figura 29* Sidebar en sistema api de reconocimiento facial. Información tomada directa de la investigación. Elaborado por Anthony Limber Chillan Pincay.

### 3.10. Resultados

#### 3.10.1. Descripción de funcionalidades

Al evaluar el todo sentido de la funcionalidad es necesario explicar explícitamente cada botón y clases que se encuentran en el Sidebar, el que cual se menciona anteriormente en el que se compone de numerosas funcionalidades.

#### 3.10.2. Reconocimiento facial único por imagen

Una de las variantes en el sistema es el reconocimiento facial mediante una imagen guardada previamente en la computadora, para un mejor funcionamiento de este.

*Figura 30 Sección de reconocimiento de rostros por imagen. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay.*

Como se puede observar en la figura 27 la vista tiene como vía principal reconocer el rostro mediante varios detectores SSD Mobilenet V1 y TinyFaceDetector, se puede usar mediante el url de una imagen o subir compatibles de imágenes como jpg, png, etc.

El reconocimiento se da de manera asíncrona con el sistema sin necesidad de recargar el navegador y muestra a la imagen subida el reconocimiento del rostro como se puede observar en la figura 31.



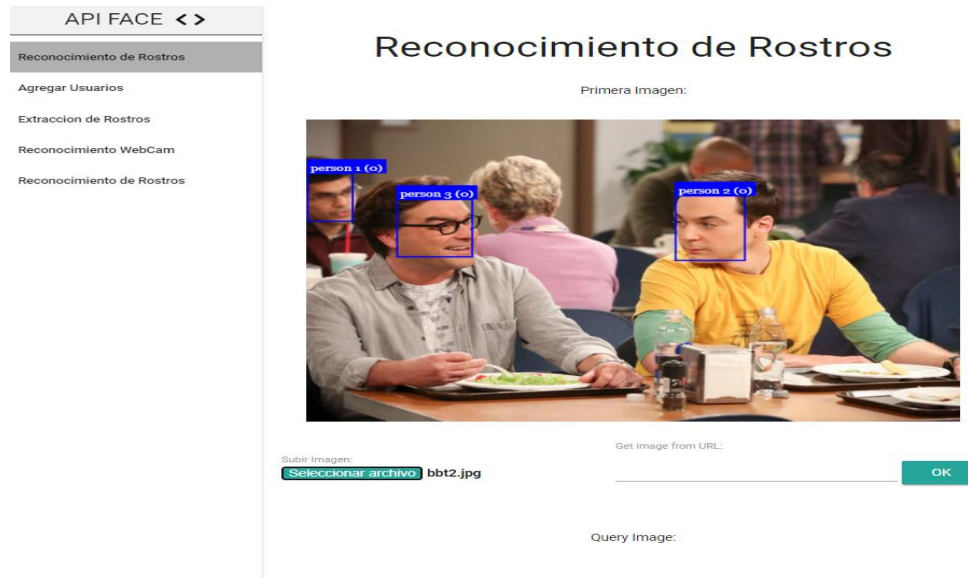


Figura 31 Rostros reconocidos por funcionalidad de imagen. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay.

### 3.10.3. Agregar usuarios

El formulario hecho para el reconocimiento previo del rostro a través de esto la funcionalidad de los demás módulos no podrían funcionar sin un registro de cada persona.

Los campos que contienen son: nombre, apellidos, y un input de imágenes en donde subes el rostro para reconocer, a través de un botón se realiza el guardado de la imagen y el resto de datos mencionados anteriormente, como se puede observar en la figura 32.

Figura 32 Formulario de registro de usuarios. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay.

### 3.10.4. Extracción de rostros

Este módulo es el principal para el reconocimiento y toma de asistencias a estudiantes, su funcionalidad es por demás la más importante en el desarrollo del API, su uso es exactamente como observa anteriormente, la subida de la imagen puede ser por imagen url, o por imagen previamente guardada en nuestro dispositivo como se ve en la figura 33.

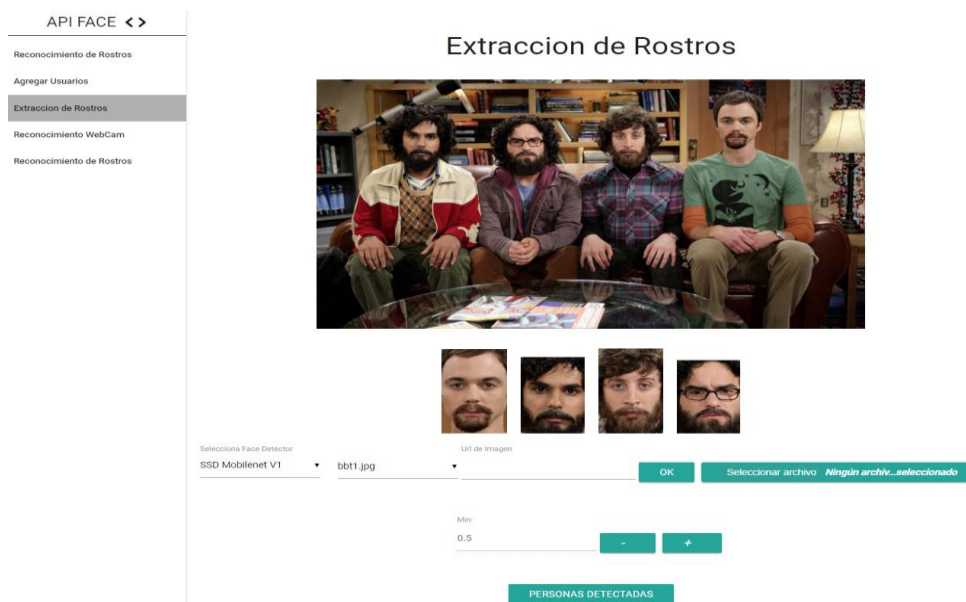


Figura 33 Extracción de rostros reconocidos API. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay.

En el módulo de extracción de rostros, se reconocen los rostros que previamente estén guardados en el sistema. El uso del botón de personas reconocidas muestra la salida de comunicación con sistemas, el formato JSON de las personas reconocidas en la extracción que cumplan con las características que definen a las personas previamente, además de la posibilidad de usar los dos detectores TinyFace Detector y MobilNet para un mejor funcionamiento en el reconocimiento.

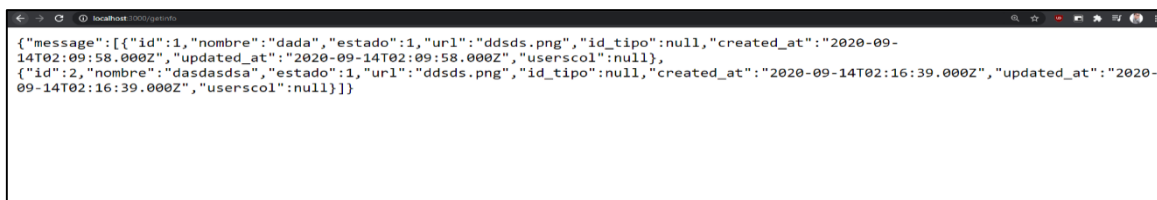


Figura 34 Salida de personas detectadas en formato JSON. Información tomada directa de la investigación. Elaborado por Anthony Limber Chilán Pincay

### **3.11. Conclusiones**

- En el presente proyecto se analizaron diferentes tecnologías del reconocimiento facial dentro de las cuales pudimos destacar Adaboost, por el uso de algoritmos más sencillos para el reconocimiento de patrones además que lo implementa OpenCv.js.
- Se verificó el uso de los recursos de software y hardware con bajos requerimientos los cuales son compatibles para el correcto funcionamiento del API con reconocimiento facial.
- Se realizó la construcción del diseño del API para el servicio de control de asistencia mediante reconocimiento facial usando librerías como NodeJs, OpenCv.js para controlar la asistencia a estudiantes, dejando un registro de las personas reconocidas como estudiantes y de personas que no se encuentren registrados mediante un log que puede ser visualizado por un usuario del API.

### **3.12. Recomendaciones**

- El uso de otro tipo de tecnologías de inteligencia artificial que mejoren el API de reconocimiento facial.
- Emplear el uso de certificado SSL por problemas de compatibilidad y seguridad de navegadores como Chrome, Firefox y Opera.
- Debido al uso de código libre en el diseño del API se podría implementar mejoras en respecto a software mediante implementación de librerías de OpenCv que mejoren el proceso de control de asistencia.

**ANEXOS**

## Anexo 1

### Manual NodeJs

```
node [options] [V8 options] [script.js | -e "script" | - ] [arguments]
```

Consulte el documento opciones de la línea de comandos para obtener más información.

#### Ejemplo

Un ejemplo de un servidor web escrito con Node.js que responde con 'Hello, World!':

Los comandos de este documento comienzan con `$o >` replican cómo aparecerían en el terminal de un usuario. No incluya los caracteres `$y >`. Están ahí para mostrar el inicio de cada comando.

Las líneas que no comienzan con un carácter `$o >` muestran el resultado del comando anterior.

Primero, asegúrese de haber descargado e instalado Node.js. Consulte Instalación de Node.js a través del administrador de paquetes para obtener más información sobre la instalación.

Ahora, cree una carpeta de proyecto vacía llamada `projects`, luego navegue hacia ella.

Linux y Mac:

```
$ mkdir ~/projects  
$ cd ~/projects
```

CMD de Windows:

```
> mkdir %USERPROFILE%\projects  
> cd %USERPROFILE%\projects
```

Windows PowerShell:

```
> mkdir $env:USERPROFILE\projects  
> cd $env:USERPROFILE\projects
```

A continuación, cree un nuevo archivo fuente en la `projects` carpeta y llámelo `hello-world.js`.

Abra `hello-world.js` en cualquier editor de texto preferido y pegue el siguiente contenido:

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello, World!\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Guarde el archivo, regrese a la ventana de la terminal e ingrese el siguiente comando:

```
$ node hello-world.js
```

Una salida como esta debería aparecer en la terminal:

```
Server running at http://127.0.0.1:3000/
```

Ahora, abra cualquier navegador web preferido y visite `http://127.0.0.1:3000`.

Si el navegador muestra la cadena `Hello, World!`, eso indica que el servidor está funcionando.

## Anexo 2

### Extractos de Leyes

---

El reconocimiento facial es un sistema o aplicación que permite identificar el rostro facial del estudiante por medio de las características de detección de rostros en mundo de la tecnología como lo es la visión artificial”, es por esto, La Unión Europea redacta que La Ley Orgánica 15/1900 establece en el (artículo 4- 14) se basa en la “protección de los datos”, en este caso biométricos, Visio computacional, entre otros.

**4. Elaboración de perfiles:** Toda forma de tratamiento automatizado de datos personales consiste en utilizar datos personales para evaluar determinados aspectos personales de una persona física, en particular para analizar o predecir aspectos relativos al rendimiento profesional, situación económica, salud, preferencias personales, intereses, fiabilidad, comportamiento, ubicación o movimiento de dicha persona física;

**Datos Biométricos:** Datos personales a partir de un tratamiento técnico específico, relativos a las características físicas, fisiológicas o conductuales de una persona física que terminan o confirman la identificación de dicha persona, como imágenes faciales o datos dactiloscópicos.

“La Ley Orgánica 1/1982, en el (artículo 9-2) del Reglamento General de protección de datos, que el responsable del fichero en su caso, el encargado del tratamiento deberá aportar las medidas de índole técnica y organizativas necesarias que garanticen la seguridad de los datos de carácter personal y eviten su alteración, pérdida, tratamiento o acceso no autorizado, habida cuenta del estado de la que están expuestos, ya provengan de la acción humana o del medio físico natural.

El mundo global en la protección del software requiere de una protección de la adquisición del programa autorizado por el propietario mediante licencia o copias de seguridad.

**La Ley de la Propiedad Intelectual** señala que el **Art. 28.-** Los programas del computador son protegidos, otorgados independientemente que son incorporados por un ordenador, demostrado por el hombre (código fuente) o por la máquina (código objeto), ya sean programas operativos y aplicativos, incluyendo diagramas de flujo, planos, manuales de uso, y en general, aquellos elementos que conformen la estructura, secuencian y organización del programa

### Anexo 3

#### Extracto de la Ley de Uso de Software Libre en el Ecuador

---

Este Software Libre en los sistemas y equipamientos informáticos dentro de la Administración Pública del Ecuador, expuesto el 10 de abril del 2008 del Decreto Ejecutivo No. 1014, que el Gobierno Ecuatoriano tiene el interés alcanzar soberanía y autonomía tecnológica, así como un ahorro de recursos públicos.

**Art. 1:** Establecer como política pública para las Entidades de la Administración Pública Central la utilización de Software Libre en sus sistemas y equipamientos informáticos.

**Art. 2:** Se entiende por Software Libre a los programas de computación que se pueden utilizar y distribuir sin restricción alguna, que permite el acceso a sus códigos fuentes y que sus aplicaciones pueden ser mejoradas.

- a) Utilización del programa con cualquier propósito de uso común.
- b) Distribución de copias sin restricciones alguna.
- c) Estudio y modificación del programa
- d) Publicación del programa mejorado.

**Art. 3:** Las entidades de la administración pública central previa a la instalación del software libre en sus equipos, deberán verificar la existencia de capacidad técnica que brinde el soporte necesario para este tipo de software

**Art. 4. -** Se faculta la utilización de software propietario (no libre) únicamente cuando no exista una solución de software libre que supla las necesidades requeridas, o cuando esté en riesgo de seguridad nacional, o cuando el proyecto informático se encuentre en un punto de no retorno.

**Art. 5. -** Tanto para software libre como software propietario, siempre y cuando se satisfagan los requerimientos.

---

*Artículo de la Ley de Uso de Software Libre en el Ecuador, Información tomada del Reglamento de la Educación Superior.*



## Anexo 4

### Extracto de la Constitución de la República del Ecuador

---

#### Educación:

---

**Art. 26.-** La educación es un derecho de las personas a lo largo de su vida y un deber ineludible e inexcusable del Estado. Constituye un área prioritaria de la política pública y de la inversión estatal, garantía de la igualdad e inclusión social y condición indispensable para el buen vivir. Las personas, las familias y la sociedad tienen el derecho y la responsabilidad de participar en el proceso educativo.

**Art. 350.-** La Constitución de la República del Ecuador señala que el Sistema de Educación Superior tiene como finalidad la formación académica y profesional con visión científica y humanista; la investigación científica y tecnológica; la innovación, promoción, desarrollo y difusión de los saberes y las culturas; la construcción de soluciones para los problemas del país, en relación con los objetivos del régimen de desarrollo.

---

#### Jóvenes

---

**Art. 39.-** El Estado garantiza los derechos de las jóvenes y los jóvenes, y promoverá su efectivo ejercicio a través de políticas y programas, instituciones y recursos que aseguren y mantengan de modo permanente su participación e inclusión en todos los ámbitos, en particular en los espacios del poder público. El Estado reconocerá a las jóvenes y los jóvenes como actores estratégicos del desarrollo del país, y les garantizará la educación, salud, vivienda, recreación, deporte, tiempo libre, libertad de expresión y asociación.

*Artículos de la Constitución de la República del Ecuador, Información tomada del Reglamento Interno la Asamblea Constitucional.*

## Anexo 5

### Extracto de Régimen del Buen Vivir

---

#### **Ciencia, tecnología, innovación y saberes ancestrales.**

**Art. 385.-** El sistema nacional de ciencia, tecnología, innovación y saberes ancestrales, en el marco del respeto al ambiente, la naturaleza, la vida, las culturas y la soberanía, tendrá como finalidad:

1. Generar, adaptar y difundir conocimientos científicos y tecnológicos.
2. Recuperar, fortalecer y potenciar los saberes ancestrales.
3. Desarrollar tecnologías e innovaciones que impulsen la producción nacional, eleven la deficiencia y productividad, mejoren la calidad de vida y contribuyan a la realización del buen vivir.

**Art. 386.-** El sistema comprenderá programas, políticas, recursos, acciones, e incorporará a instituciones del Estado, universidades y escuelas politécnicas, institutos de investigación públicos y particulares, empresas públicas y privadas, organismos no gubernamentales y personas naturales o jurídicas, en tanto realizan actividades de investigación, desarrollo tecnológico, innovación y aquellas ligadas a los saberes ancestrales.

---

*Artículos del Régimen del Buen Vivir, Información tomada de la investigación previa de innovación y saberes ancestrales.*

## Anexo 6

### Extracto de Ley Orgánica de Educación Superior

---

#### **Título I - Ámbito, Objeto, Fines y Principios del Sistema de Educación Superior.**

---

**Art. 5.-** Derechos de las y los estudiantes. - Son derechos de las y los estudiantes los siguientes ítems:

a) Acceder, movilizarse, permanecer, egresar y titularse sin discriminación conforme sus méritos académicos;

i) Obtener de acuerdo con sus méritos académicos becas, créditos y otras formas de apoyo económico en el proceso de formación de educación superior.

**Art. 6.-** Derechos de los profesores o profesoras e investigadores o investigadoras. – Son derechos de los profesores o profesoras e investigadores o investigadoras de conformidad con la constitución y esta Ley los siguientes: Acceder a la carrera de profesor e investigador y a cargos directivos, que garantice estabilidad, promoción, movilidad y retiro, basados en el mérito académico, en la calidad de la enseñanza impartida, en la producción investigativa, en el perfeccionamiento permanente, sin admitir discriminación de género ni de ningún otro tipo.

---

*Artículo de la Ley Orgánica de la Educación Superior, Información tomada del Reglamento de la Educación Superior.*

## Anexo 7

### Código Orgánico Integral Penal

---

**Art. 234.-** “Acceso no consentido a un sistema informático, telemático o de telecomunicaciones.- La persona que sin autorización acceda en todo o en parte a un sistema informático o sistema telemático o de telecomunicaciones o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho, para explotar ilegítimamente el acceso logrado, modificar un portal web, desviar o re direccionar de tráfico de datos o voz u ofrecer servicios que estos sistemas proveen a terceros, sin pagarlos a los proveedores de servicios legítimos, será sancionada con la pena privativa de la libertad de tres a cinco años”, (COIP, 2014).

---

*Información tomada del Artículo del Código Orgánico Integral Penal.*

## Anexo 8

### Modelo de Encuesta

Diseño API con reconocimiento facial para el control de asistencia a estudiantes.

La siguiente encuesta está orientada a determinar los requerimientos, sobre los cuales se dará el diseño de un API que implemente reconocimiento facial para el control de asistencia de estudiantes, como parte de investigación de trabajo de titulación. Sabiendo que un API se trata de un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones, permitiendo la comunicación entre dos aplicaciones de software a través de un conjunto de reglas.

1. ¿Qué método usa usted frecuentemente para la toma de asistencia a estudiantes?
  - ☐ Excel
  - ☐ Sistemas Integrados en plataforma educativa
  - ☐ Bloc de Notas
  - ☐ Otros
2. ¿Considera usted eficiente el proceso de detección de rostros para el control de datos de una persona? \*
  - ☐ Totalmente de acuerdo
  - ☐ De acuerdo
  - ☐ Neutral
  - ☐ En desacuerdo
  - ☐ Totalmente en desacuerdo
3. ¿Tiene usted conocimiento sobre visión artificial para el proceso de detección de rostros? \*
  - ☐ Mucho
  - ☐ Poco
  - ☐ Nada
4. ¿Conoce usted una aplicación que implemente reconocimiento de rostros? \*

Marca solo un óvalo.

  - ☐ Si
  - ☐ No
5. ¿Tiene usted conocimiento sobre el uso de un API para detección de rostros? \*
  - ☐ Mucho
  - ☐ Poco
  - ☐ Nada
6. ¿Actualmente los docentes cuentan con una herramienta de reconocimiento de rostros para el control de asistencia a estudiantes? \*
  - ☐ Si
  - ☐ No
7. ¿Le gustaría a usted usar un API de reconocimiento facial para el uso y control de asistencia a estudiantes? \*
  - ☐ Totalmente de acuerdo
  - ☐ De acuerdo
  - ☐ Neutral
  - ☐ En desacuerdo
  - ☐ Totalmente en desacuerdo

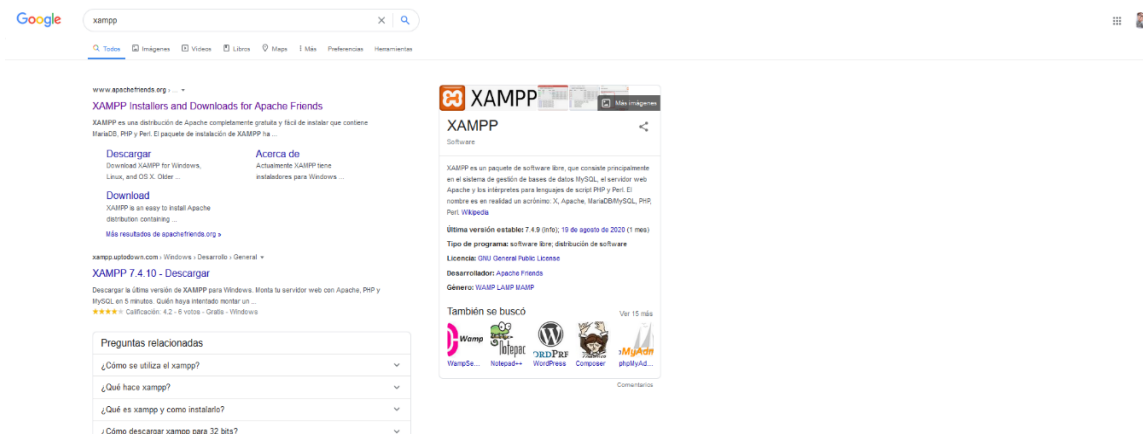
8. ¿Considera usted que el uso de esta herramienta tecnológica les servirá a los docentes para llevar un mejor control de asistencia a estudiantes? \*

- ☐ Totalmente de acuerdo
- ☐ De acuerdo
- ☐ Neutral
- ☐ En desacuerdo
- ☐ Totalmente en desacuerdo

## Anexo 9

### Instalación de paquetes de servidores XAMPP

1. Se comienza con la instalación de XAMPP mediante la página oficial.



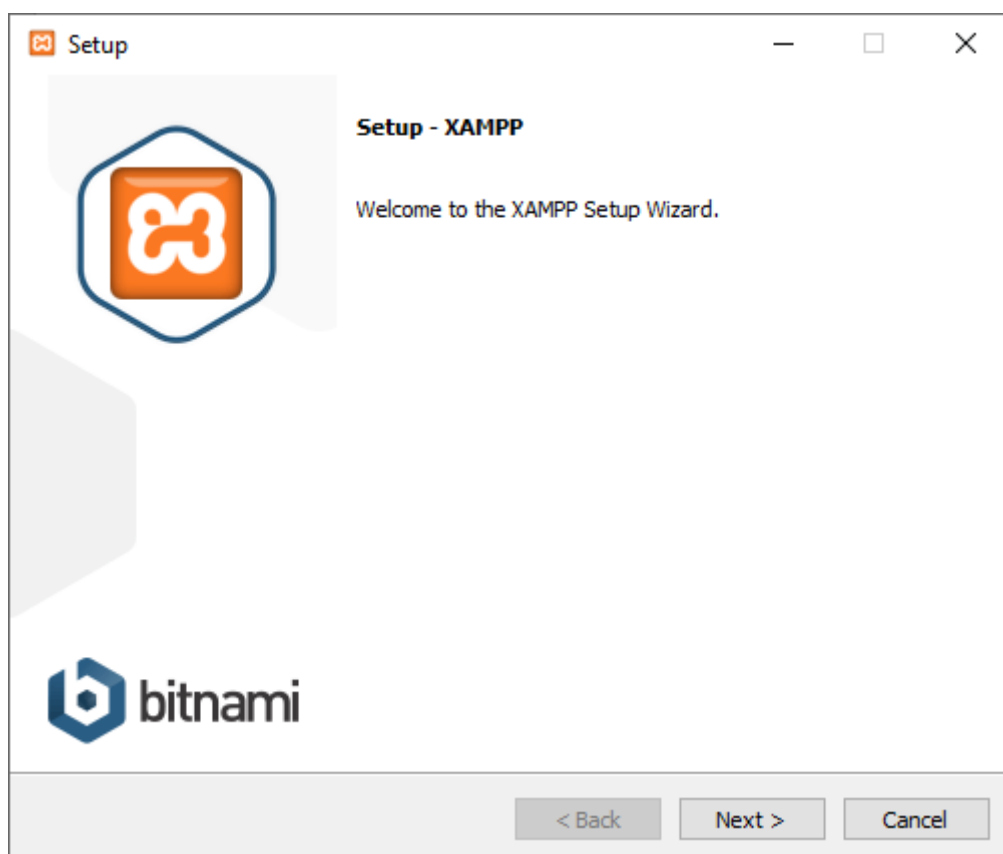
2. Se le da al botón de descargar según el sistema operativo que se esté usando.



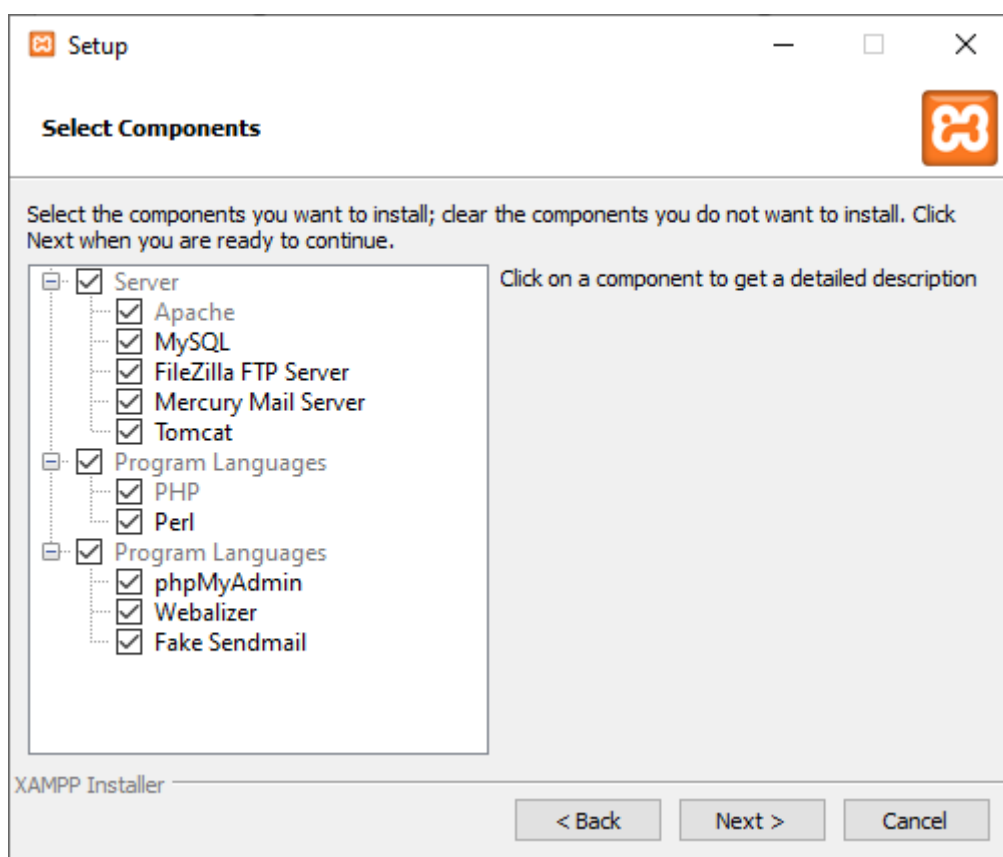
3. Luego del click guardar el archivo por preferencia en nuestro equipo.



4. Se comienza con la instalación (En este caso es Windows 10)

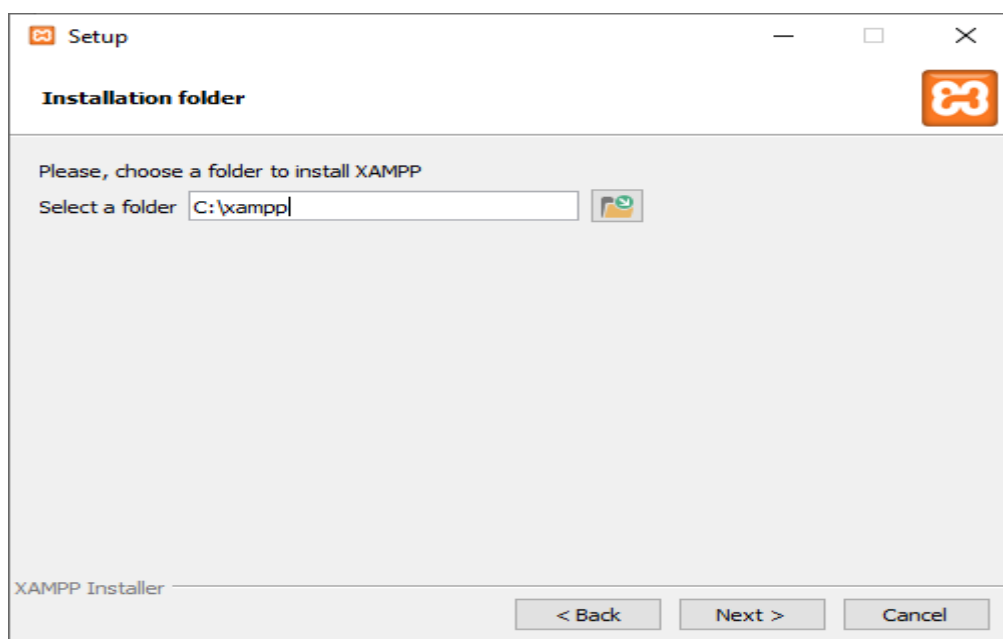


5. Se selecciona los componentes a utilizar.



6. Selecciona la carpeta donde se guarde y listo.





## Anexo 10

### Conexión con la base de datos mediante NodeJs

```
var con = mysql.createConnection({  
  host: "localhost",  
  user: "root",  
  password: "",  
  database: "faceapi"  
});
```

## Anexo 11

### Levantamiento de servidor en puerto 3000

```
const express = require('express')
const path = require('path')
const { get } = require('request')
const fileUpload = require('express-fileupload')
const app = express()

app.use(express.json())
app.use(express.urlencoded({ extended: true }))
app.use(fileUpload())
const viewsDir = path.join(__dirname, 'views')
app.use(express.static(viewsDir))
app.use(express.static(path.join(__dirname, './public')))
app.use(express.static(path.join(__dirname, '../images')))
app.use(express.static(path.join(__dirname, '../media')))
app.use(express.static(path.join(__dirname, '../..../weights')))
app.use(express.static(path.join(__dirname, '../..../dist')))

app.get('/', (req, res) => res.redirect('/face_extraction'))
app.get('/create_users', (req, res) => res.sendFile(path.join(viewsDir, 'Ad
dUserFaceDetection.html')))
app.get('/face_detection', (req, res) => res.sendFile(path.join(viewsDir, '
faceDetection.html')))
app.get('/face_landmark_detection', (req, res) => res.sendFile(path.join(vi
ewsDir, 'faceLandmarkDetection.html')))
app.get('/face_expression_recognition', (req, res) => res.sendFile(path.joi
n(viewsDir, 'faceExpressionRecognition.html')))
app.get('/age_and_gender_recognition', (req, res) => res.sendFile(path.join
(viewsDir, 'ageAndGenderRecognition.html')))
app.get('/face_extraction', (req, res) => res.sendFile(path.join(viewsDir,
'faceExtraction.html')))
app.get('/face_recognition', (req, res) => res.sendFile(path.join(viewsDir,
'faceRecognition.html')))
app.get('/video_face_tracking', (req, res) => res.sendFile(path.join(viewsD
ir, 'videoFaceTracking.html')))
app.get('/webcam_face_detection', (req, res) => res.sendFile(path.join(view
sDir, 'webcamFaceDetection.html')))
app.get('/webcam_face_landmark_detection', (req, res) => res.sendFile(path.
join(viewsDir, 'webcamFaceLandmarkDetection.html')))
app.get('/webcam_face_expression_recognition', (req, res) => res.sendFile(p
ath.join(viewsDir, 'webcamFaceExpressionRecognition.html')))
app.get('/webcam_age_and_gender_recognition', (req, res) => res.sendFile(pa
th.join(viewsDir, 'webcamAgeAndGenderRecognition.html')))
```

```

app.get('/bbt_face_landmark_detection', (req, res) => res.sendFile(path.join(viewsDir, 'bbtFaceLandmarkDetection.html')))
app.get('/bbt_face_similarity', (req, res) => res.sendFile(path.join(viewsDir, 'bbtFaceSimilarity.html')))
app.get('/bbt_face_matching', (req, res) => res.sendFile(path.join(viewsDir, 'bbtFaceMatching.html')))
app.get('/bbt_face_recognition', (req, res) => res.sendFile(path.join(viewsDir, 'bbtFaceRecognition.html')))
app.get('/batch_face_landmarks', (req, res) => res.sendFile(path.join(viewsDir, 'batchFaceLandmarks.html')))
app.get('/batch_face_recognition', (req, res) => res.sendFile(path.join(viewsDir, 'batchFaceRecognition.html')))
const multer = require('multer')({
  dest: 'public/files'
})
var bodyParser = require('body-parser');

// parse application/x-www-form-urlencoded
app.use(bodyParser.urlencoded({ extended: false }));

// parse application/json
app.use(bodyParser.json());
var mysql = require('mysql');

app.post('/uploadf', (req, res) => {
  let EDFile = req.files.file
  console.log(req.body.nombre);
  var nombre = req.body.nombre;
  var con = mysql.createConnection({
    host: "localhost",
    user: "root",
    password: "",
    database: "faceapi"
  });

  con.connect(function(err) {
    if (err) throw err;
    console.log("Conectado!");
    console.log(req.body.nombre+"aquí");
    var sql = "INSERT INTO users (nombre, url, estado) VALUES ('"+nombre+"', '"+EDFile.name+"', '1')";
    con.query(sql, function (err, result) {
      if (err) throw err;
      console.log("1 record inserted");
    });
  });

  EDFile.mv(`./files/${EDFile.name}`, err => {

```

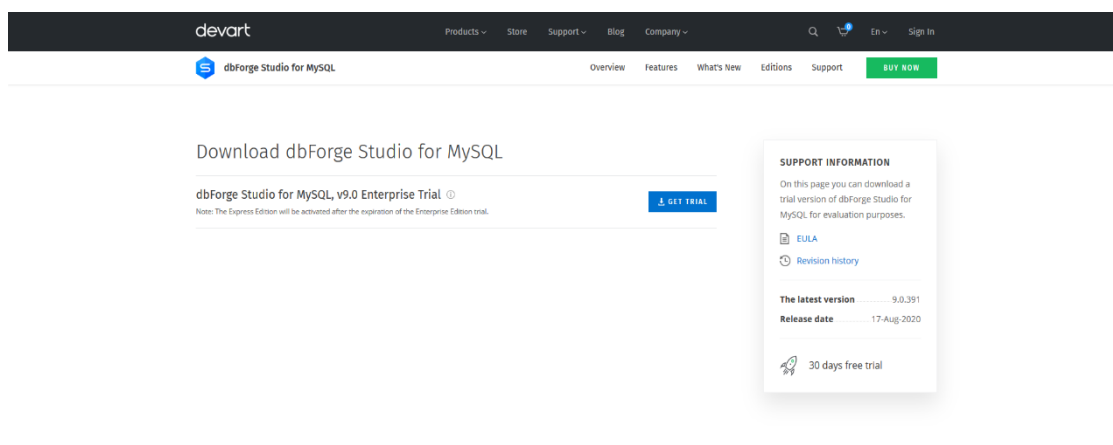
```
        if(err) return res.status(500).send({ message : err })
        //alert("Guardado Correcto");
        return res.redirect('/create_users');
    })
})
app.get('/getinfo',(req,res) => {
    var con = mysql.createConnection({
        host: "localhost",
        user: "root",
        password: "",
        database: "faceapi"
    });
    con.connect(function(err) {
        if (err) throw err;
        con.query("SELECT * FROM users", function (err, result, fields) {
            if (err) throw err;
            return res.status(500).send({ message : result })
        });
    });
})

app.listen(3000, () => console.log('Escucha del Puerto 3000!'))
```

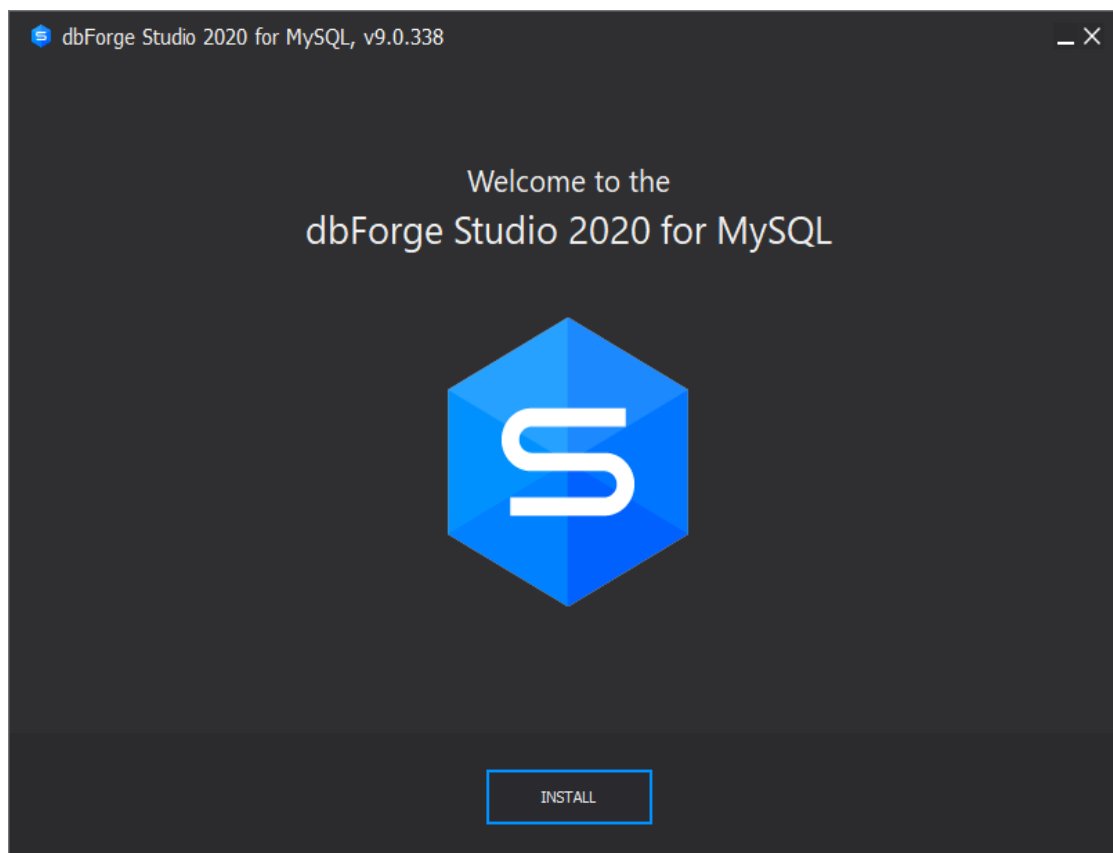
## Anexo 12

### Instalación de dbForge Studio

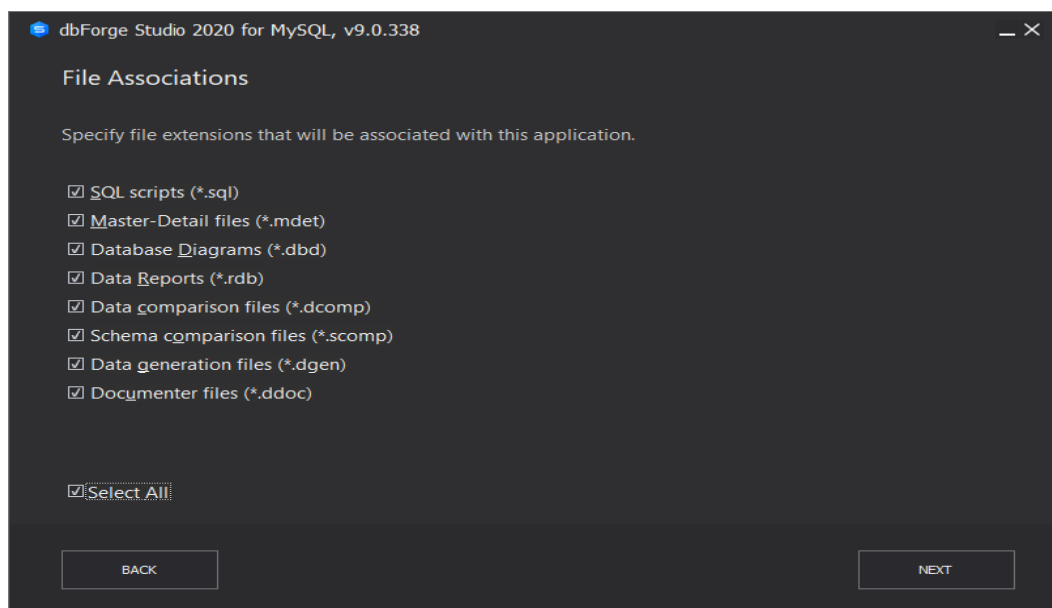
1. Descargar dbForge Studio para MySQL según nuestro sistema operativo



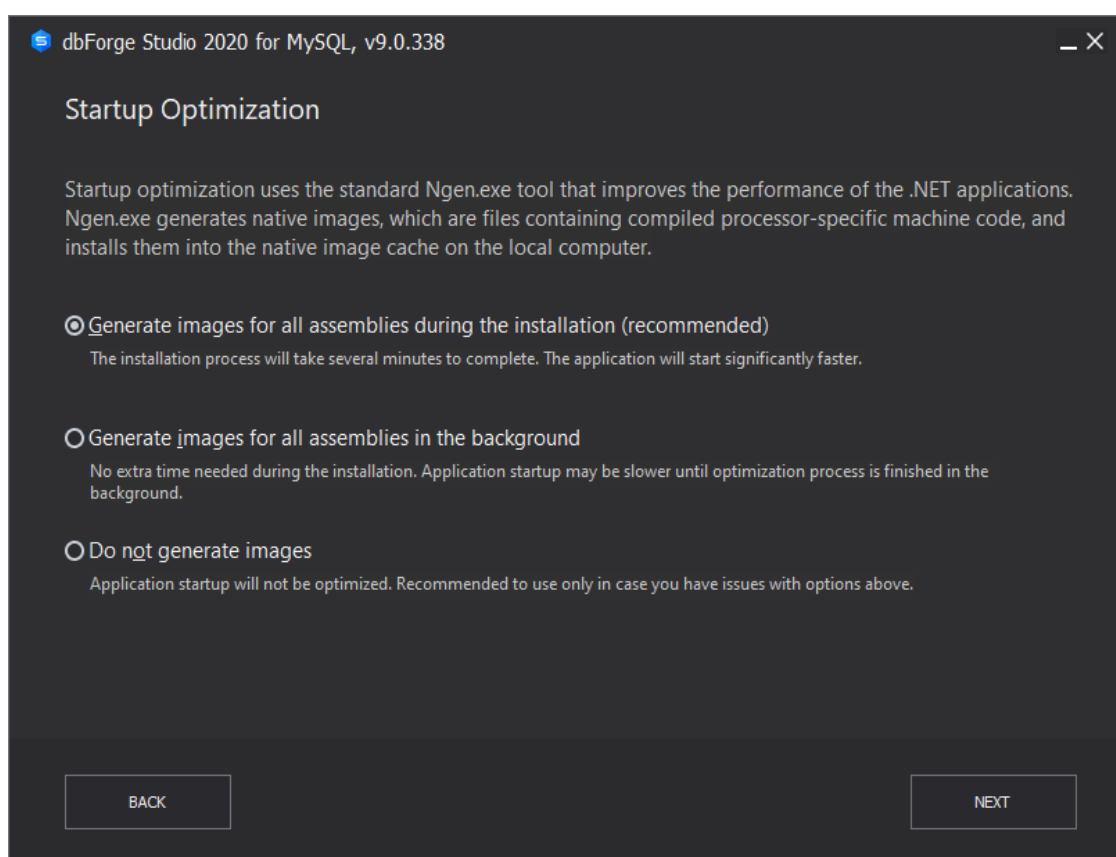
2. Aparece ventana de instalación y siguiente.



3. Instalar todas las dependencias que requiere MySQL



4. Se escoge la opción de recomendada por el instalador de dbForge y proceder con la instalación.



## Bibliografía

- Tarazona Ospina, I., & MSc. Daniel, V. (2015). RECONOCIMIENTO FACIAL BASADO EN EIGENFACES, LBHP Y FISHERFACES EN LA BEAGLEBOARD-xM. *Revista colombiana de tecnologías de avanzada*, 148,149.
- Alvarez, M. A. (19 de Noviembre de 2015). *Desarrolloweb.com*. Obtenido de ¿Que es python?: <https://desarrolloweb.com/articulos/1325.php>
- BBVAOPEN4U. (23 de Marzo de 2016). *BBVAOPEN4U*. Obtenido de <https://bbvaopen4u.com/es/actualidad/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos>
- COIP. (2014). *Libro de los Artículos del Código Orgánico Penal*. Obtenido de [https://tbinternet.ohchr.org/Treaties/CEDAW/Shared%20Documents/ECU/INT\\_CEDAW\\_ARL\\_ECU\\_18950\\_S.pdf](https://tbinternet.ohchr.org/Treaties/CEDAW/Shared%20Documents/ECU/INT_CEDAW_ARL_ECU_18950_S.pdf)
- E3FF. (2 de Enero de 2014). *E3FF*. Obtenido de <https://www.eff.org/es/pages/face-recognition>
- Emscripten. (2015). *Emscripten*. Obtenido de <https://emscripten.org/>
- Enatec. (24 de Mayo de 2010). *Enatec*. Obtenido de <https://einatec.com/control-acceso-biometrico/>
- Gebhart, A. (27 de Marzo de 2019). *Cnet*. Obtenido de Cnet: <https://www.cnet.com/es/noticias/reconocimiento-facial-apple-amazon-google-ai/>
- Gendra, M. (4 de Mayo de 2010). <https://www.marianogendra.com.ar/Articulos/que-es-una-api>. Obtenido de <https://www.marianogendra.com.ar/Articulos/que-es-una-api>
- Martinez, N. (4 de Junio de 2019). *IEBS*. Obtenido de [https://www.iebschool.com/blog/reconocimiento-facial-tecnologia/#%C2%BFQue\\_es\\_el\\_reconocimiento\\_facial](https://www.iebschool.com/blog/reconocimiento-facial-tecnologia/#%C2%BFQue_es_el_reconocimiento_facial)



- Maza, E. A., & Goyes Mosquera, J. J. (2015). *Repositorio Digital*. Obtenido de Sistema de información y control de asistencia del personal para Facultad de Ingeniería.: <http://www.dspace.uce.edu.ec/handle/25000/496>
- MELO, S. (2 de Octubre de 2019). *DataScope*. Obtenido de <https://mydatascope.com/blog/es/api-integrations-5-ways-they-can-benefit-your-business-2/>
- OpenCv. (2016). *OpenCv*. Obtenido de [https://docs.opencv.org/3.4/df/d0a/tutorial\\_js\\_intro.html](https://docs.opencv.org/3.4/df/d0a/tutorial_js_intro.html)
- Ottado, G. (2015). *Reconocimiento de caras: Eigenfaces y Fisherfaces*. Uruguay: Universidad de ORT.
- Redondo, M. (22 de Junio de 2018). *Erre que Erre*. Obtenido de <http://www.errequeerre.es/anatomia-de-una-peticion-a-una-api-rest>
- Robledano, Á. (24 de Septiembre de 2019). *OpenWebinars*. Obtenido de <https://openwebinars.net/blog/que-es-mysql/>
- Slate, A. (8 de Julio de 2019). *Wrike*. Obtenido de <https://www.wrike.com/es/blog/que-es-una-api-necesitas-saber/>
- Therneau, T. a. (2019). *The Comprehensive R Archive Network*. Obtenido de <https://cran.r-project.org/web/packages/adabag/index.html>
- Viola , P., & Jones, M. (16 de Enero de 2001). *Carnegie Mellon University*. Obtenido de <https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf>