



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES DE LOS DOCUMENTOS
PERSONALES DE IDENTIFICACIÓN DE GUATEMALA, USANDO INTELIGENCIA
ARTIFICIAL EN LA NUBE**

José Luis Herrera Martínez

Asesorado por Ing. Erick Carlos Roberto Navarro Delgado

Guatemala mayo de 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES DE LOS DOCUMENTOS
PERSONALES DE IDENTIFICACIÓN DE GUATEMALA, USANDO INTELIGENCIA
ARTIFICIAL EN LA NUBE**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

JOSÉ LUIS HERRERA MARTÍNEZ

ASESORADO POR EL ING. ERICK CARLOS ROBERTO NAVARRO
DELGADO

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, MAYO DE 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton De León Bran
VOCAL IV	Br. Kevin Vladimir Armando Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANA	Inga. Aurelia Anabela Cordova Estrada
EXAMINADOR	Ing. Pedro Pablo Hernández Ramírez
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
EXAMINADOR	Ing. Carlos Alfredo Azurdia Morales
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES DE LOS DOCUMENTOS
PERSONALES DE IDENTIFICACIÓN DE GUATEMALA, USANDO INTELIGENCIA
ARTIFICIAL EN LA NUBE**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas con fecha 01 de junio de 2022.

A handwritten signature in black ink, appearing to read 'José Luis Herrera Martínez', written in a cursive style.

José Luis Herrera Martínez

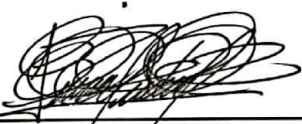
Guatemala, 21 de marzo de 2023

Ingeniero
Carlos Alfredo Azurdia
Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Respetable Ingeniero Azurdia:

Por este medio hago de su conocimiento, que como asesor del trabajo de graduación del estudiante de la carrera de Ingeniería en Ciencias y Sistemas, **José Luis Herrera Martínez**, quien se identifica con el código único de identificación **3006 76298 0101** y con el registro académico **201807431**, hago constar que ha finalizado todos los capítulos del trabajo de investigación titulado: **"Sistema de reconocimiento óptico de caracteres de los documentos personales de identificación de Guatemala, usando inteligencia artificial en la nube"**, el cual he tenido la oportunidad de revisar y doy mi aprobación al mismo.

Agradeciendo su atención a la presente, me es grato suscribirme.

F: 
Ing. Erick Carlos Roberto Navarro Delgado
Colegiado No. 16465

Erick Carlos Roberto Navarro Delgado
Ingeniero en Ciencias y Sistemas
No. Colegiado: 16465



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala 30 de marzo de 2023

Ingeniero
Carlos Gustavo Alonzo
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Alonzo:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **JOSÉ LUIS HERRERA MARTÍNEZ** con carné **201807431** y CUI **3006 76298 0101** titulado “**SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES DE LOS DOCUMENTOS PERSONALES DE IDENTIFICACIÓN DE GUATEMALA, USANDO INTELIGENCIA ARTIFICIAL EN LA NUBE**”, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo aprobado.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,



Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



SIST.LNG.DIRECTOR.2.EICCSS.2023

El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del Asesor, el visto bueno del Coordinador de área y la aprobación del área de lingüística del trabajo de graduación titulado: **SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES DE LOS DOCUMENTOS PERSONALES DE IDENTIFICACIÓN DE GUATEMALA, USANDO INTELIGENCIA ARTIFICIAL EN LA NUBE**, presentado por: **José Luis Herrera Martínez**, procedo con el Aval del mismo, ya que cumple con los requisitos normados por la Facultad de Ingeniería.

“ID Y ENSEÑAD A TODOS”



Ingeniero Carlos Gustavo Alonzo
DIRECTOR
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, mayo de 2023



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala

Decanato
Facultad e Ingeniería

24189101- 24189102

LNG.DECANATO.OIE.5.2023

La Decana de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería En Ciencias Y Sistemas, al Trabajo de Graduación titulado: **SISTEMA DE RECONOCIMIENTO ÓPTICO DE CARACTERES DE LOS DOCUMENTOS PERSONALES DE IDENTIFICACIÓN DE GUATEMALA, USANDO INTELIGENCIA ARTIFICIAL EN LA NUBE**, presentado por: **José Luis Herrera Martínez** después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:

Firmado digitalmente por AURELIA ANABELA CORDOVA ESTRADA Fecha: 14/05/2023 11:31:05 p.m. Razón: Orden de impresión Ubicación: Facultad de Ingeniería, USAC.

Inga. Aurelia Anabela Córdoba Estrada
Decana



Guatemala, mayo de 2023

Para verificar validez de documento ingrese a <https://www.ingenieria.usac.edu.gt/firma-electronica/consultar-documento>

Tipo de documento: Correlativo para orden de impresión Año: 2023 Correlativo: 5 CUI: 3006762980101

Escuelas: Ingeniería Civil, Ingeniería Mecánica Industrial, Ingeniería Química, Ingeniería Mecánica Eléctrica, - Escuela de Ciencias, Regional de Ingeniería Sanitaria y Recursos Hidráulicos (ERIS). Postgrado Maestría en Sistemas Mención Ingeniería Vial. Carreras: Ingeniería Mecánica, Ingeniería Electrónica, Ingeniería en Ciencias y Sistemas. Licenciatura en Matemática. Licenciatura en Física. Centro de Estudios Superiores de Energía y Minas (CESEM). Guatemala, Ciudad

ACTO QUE DEDICO A:

Dios

Quien me dio el don del aprendizaje. Espero que este trabajo honre tu sabiduría y sea un reflejo de tu amor por la creación.

Mi padre

Quien siempre estará presente en mi corazón y en mi memoria. Gracias por enseñarme el valor del esfuerzo y la dedicación. Esta tesis es un tributo a tu ejemplo de vida y a tu legado.

A la memoria de mi querido padre, quien me inspiró a perseguir mis sueños y a nunca rendirme. Espero que desde donde estés, puedas ver este logro y sentirte orgulloso de mí. (q. e. p. d.).

Mis hermanos

Quienes han sido mi apoyo constante en cada etapa de mi vida. Gracias por su amor incondicional, su paciencia y su motivación. Esta tesis es también para ustedes, que han sido una parte fundamental de mi camino hacia el éxito.

AGRADECIMIENTOS A:

- Universidad de San Carlos de Guatemala** Por ser mi casa de estudios y permitirme desarrollar mis habilidades y mis capacidades que son de gran valor para la sociedad.
- Mi padre** Gracias por darme la fuerza y la motivación para poder lograr mis sueños y metas, gracias por enseñarme a valorar y a esforzarme cada día.
- Mis amigos** Con los cuales pasé grandes momentos dentro de la universidad, gracias por los momentos amenos y por siempre estar en los momentos más difíciles.
- Mi asesor** Ing. Erick Carlos Roberto Navarro Delgado, gracias por permitirme ser su asesorado y brindarme de su conocimiento para la elaboración de este trabajo.
- Mi madre** Gracias por permitirme poder estudiar y desarrollarme como persona y como profesional, por el esfuerzo para poder seguir apoyándome en mis sueños y metas.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
GLOSARIO	VII
RESUMEN.....	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN	XV
1. MARCO TEÓRICO.....	1
1.1. Introducción a la inteligencia artificial y su aplicación en el reconocimiento de imágenes y caracteres.....	1
1.2. Descripción de la nube utilizada para la construcción del sistema.....	2
1.2.1. Azure	2
1.3. Reconocimiento óptico de caracteres (OCR).	3
1.4. Documentos personales de identificación de Guatemala y sus características.	5
1.4.1. Documento personal de identificación (DPI).....	5
1.4.2. Licencia de conducir	7
2. IDENTIFICACIÓN DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN..	11
2.1. Antecedentes.....	11
2.2. Descripción del problema	12
2.3. Marcado objetivo	13
3. DOCUMENTO DE ESPECIFICACIONES, REQUERIMIENTOS Y CRITERIOS DE ACEPTACIÓN.....	15

3.1.	Requerimientos	15
3.1.1.	Requerimientos funcionales	15
3.1.2.	Requerimientos no funcionales	16
3.2.	Alcances.....	16
3.3.	Limitantes.....	17
3.4.	Arquitectura de la solución	18
3.5.	Diagrama de flujo de la solución	19
4.	DISEÑO DE LA APLICACIÓN	21
4.1.	Descripción del software	21
4.2.	Diseño de la aplicación web.....	22
4.2.1.	Vistas.....	22
4.2.1.1.	Página Home.....	22
4.2.1.2.	Página OCR DPI	23
4.2.1.3.	Página OCR Licencia de conducir.....	24
4.3.	Diseño del API	24
4.3.1.	Endpoints	25
4.3.1.1.	Datos del DPI	25
4.3.1.2.	Datos de la licencia de conducir	26
4.3.1.3.	Obtener todos los registros	27
4.4.	Datos de entrenamiento	28
4.5.	Herramientas y tecnologías utilizadas en el desarrollo del sistema	30
4.5.1.	Detección de objetos de Azure.....	30
4.5.2.	Angular	31
4.5.3.	Node JS.....	32
4.5.4.	Express JS	33
4.5.5.	Tesseractjs	34
4.5.6.	MongoDB	35

4.5.7.	Visual Studio Code	36
5.	EVALUACIÓN DEL SISTEMA.....	39
5.1.	Análisis de resultados obtenidos	39
	CONCLUSIONES	43
	RECOMENDACIONES	45
	REFERENCIAS	47
	APÉNDICES	49

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	DPI parte frontal.....	6
2.	DPI parte trasera	7
3.	Licencia de conducir de Guatemala parte frontal	9
4.	Licencia de conducir de Guatemala parte trasera	10
5.	Arquitectura de la solución	19
6.	Diagrama de flujo de la solución	20
7.	Página Home de la aplicación web	23
8.	Página OCR DPI de la aplicación web	23
9.	Página OCR Licencia de la aplicación web	24
10.	Ejemplo de request para el endpoint datos del dpi.....	25
11.	Ejemplo de response para el endpoint datos del dpi.....	26
12.	Ejemplo de request para el endpoint datos de la licencia de conducir ...	26
13.	Ejemplo de response para el endpoint datos de la licencia de conducir.....	27
14.	Ejemplo de response para el endpoint obtener todos los datos	28
15.	Datos de entremiento DPI Custom Vision Azure.....	29
16.	Datos de entrenamiento Licencia Custom Vision Azure.....	30
17.	Resultados obtenidos Custom Vision Azure DPI.....	40
18.	Resultados obtenidos Custom Vision Azure Licencia de conducir	41

GLOSARIO

API	Es un conjunto de reglas y protocolos que permiten a los programas de software interactuar entre sí. Define los métodos y formatos de intercambio de datos entre aplicaciones y permite a los desarrolladores integrar las funcionalidades de diferentes sistemas o servicios en sus propias aplicaciones.
Aplicación Web	Es una aplicación de software accesible por medio de un navegador web, y que se ejecuta en un servidor remoto. Las aplicaciones web permiten a los usuarios interactuar con ellas a través de una interfaz web, y pueden proporcionar una amplia gama de funcionalidades, como procesamiento de formularios, acceso a bases de datos y servicios de mensajería.
Bytes	Es la unidad fundamental de medida de la información digital y representan un conjunto de ocho bits (cada bit es un dígito binario que puede ser 0 o 1). Los bytes se utilizan comúnmente para almacenar y transmitir información digital, como archivos de computadora, imágenes y sonidos.
Cliente	Es un software que se utiliza para acceder y mostrar información y servicios disponibles en la World Wide Web. Ejemplos comunes de clientes web incluyen

navegadores web como Google Chrome, Mozilla Firefox y Microsoft Edge. Los clientes web solicitan y reciben información de los servidores web por medio del protocolo HTTP o HTTPS.

Endpoint

Es un punto final de una comunicación entre dos sistemas o servicios. En el contexto de las APIs, un endpoint se refiere a una URL específica que puede ser utilizada por una aplicación para acceder a un recurso o funcionalidad proporcionada por un servicio web.

Form-Data

Es un formato de datos utilizado en la transmisión de información a través de formularios web. Este formato se utiliza para enviar datos de formulario, como texto y archivos, desde el cliente (como un navegador web), al servidor web.

Inteligencia Artificial

Es un conjunto de tecnologías y técnicas que permiten a las computadoras realizar tareas que requieren inteligencia humana, como el aprendizaje, la percepción, el razonamiento y la resolución de problemas. La IA utiliza algoritmos y modelos matemáticos para analizar y procesar datos y tomar decisiones.

JSON

Es un formato de intercambio de datos ligero y fácil de leer y escribir que se utiliza comúnmente en

aplicaciones web para transmitir y almacenar datos estructurados.

Nube

Es un término utilizado para referirse a la entrega de servicios de computación a través de internet, donde los usuarios pueden acceder a recursos informáticos compartidos, como servidores, almacenamiento y aplicaciones, desde cualquier lugar y en cualquier momento.

OCR

Es una tecnología que permite convertir imágenes de texto impreso o manuscrito en texto digital editable mediante la detección de patrones de texto en la imagen.

Request

Es una solicitud enviada por un cliente a un servidor para obtener una respuesta o realizar una acción. En el contexto de las aplicaciones web y las APIs, un request puede incluir datos de entrada, como parámetros de consulta o datos de formulario, que se utilizan para procesar la solicitud y generar una respuesta adecuada.

Response

Es una respuesta enviada por un servidor a un cliente después de recibir una solicitud (request). En el contexto de las aplicaciones web y las APIs, un response incluye los datos de salida generados por el servidor en respuesta a la solicitud, como HTML, JSON, XML, entre otros.

Servidor

Es un software o un sistema de hardware que proporciona servicios y recursos a otros dispositivos y programas conectados a una red. Los servidores se utilizan comúnmente para almacenar y administrar datos, alojar sitios web, ejecutar aplicaciones y servicios, y permitir el intercambio de información entre dispositivos en una red.

Sistema

Es un conjunto de elementos interrelacionados que trabajan juntos para cumplir un objetivo o función específica. En informática, un sistema se refiere a un conjunto de programas, procesos y recursos que trabajan juntos para realizar una tarea determinada.

RESUMEN

La tesis se enfoca en el desarrollo de un sistema de reconocimiento óptico de caracteres (OCR), para los documentos personales de identificación de Guatemala utilizando inteligencia artificial en la nube.

El sistema propuesto permitirá la verificación automatizada de la identidad y reducirá el tiempo y el costo asociados con la verificación manual. Se realizará una revisión de la literatura existente sobre el reconocimiento óptico de caracteres y la inteligencia artificial en la nube, se propondrá una solución técnica para el reconocimiento automático de la información contenida en los documentos.

El rendimiento del sistema propuesto será evaluado exhaustivamente utilizando un conjunto de datos de documentos de identificación personal de Guatemala.

El resultado esperado es un sistema más preciso y eficiente para la verificación de la identidad en Guatemala, que pueda proporcionar una solución efectiva y rentable para mejorar el proceso de verificación de la identidad.

OBJETIVOS

General

Desarrollar un sistema de reconocimiento óptico de caracteres (OCR), Para los documentos de identificación personales de Guatemala utilizando inteligencia artificial en la nube para reducir el tiempo y costo asociados a la digitalización del documento de identificación manual.

Específicos

1. Aplicar la tecnología de reconocimiento óptico de caracteres (OCR), para convertir una imagen del documento personal de identificación y licencia de conducir de Guatemala en un texto digital, como parte del proceso de verificación de la identidad.
2. Realizar una investigación sobre las técnicas y algoritmos de reconocimiento óptico de caracteres (OCR), y aprendizaje automático utilizados en la industria para seleccionar el algoritmo más eficiente y confiable para el reconocimiento oficial de identificaciones personales.
3. Entrenar y ajustar el modelo de inteligencia artificial (IA), con un conjunto de datos de documentos de identificación personal de Guatemala hasta alcanzar una precisión superior al 85 %.

INTRODUCCIÓN

La verificación de la identidad es un proceso crucial en una amplia variedad de transacciones y servicios en Guatemala, como la apertura de cuentas bancarias y la emisión de pasaportes, entre otros. Actualmente, este proceso se realiza de manera manual, lo que puede ser costoso y lento.

En este contexto, el uso de tecnologías como el reconocimiento óptico de caracteres (OCR), y la inteligencia artificial en la nube pueden ofrecer una solución efectiva para mejorar el proceso de verificación de la identidad de los documentos personales de identificación en Guatemala. El OCR permite la extracción automática de información de los documentos, mientras que la inteligencia artificial en la nube permite el procesamiento rápido y eficiente de grandes cantidades de datos.

El objetivo de esta tesis es desarrollar un sistema de reconocimiento óptico de caracteres para los documentos personales de identificación de Guatemala mediante el uso de inteligencia artificial en la nube.

Este sistema permitirá la verificación automatizada de la identidad, reduciendo así el tiempo y el costo asociados con la verificación manual. Se espera que el sistema de OCR propuesto sea preciso y eficiente en su funcionamiento.

1. MARCO TEÓRICO

1.1. Introducción a la inteligencia artificial y su aplicación en el reconocimiento de imágenes y caracteres

La inteligencia artificial (IA), es una disciplina de la informática que se centra en el desarrollo de sistemas que pueden realizar tareas que requieren inteligencia humana, como el razonamiento, la percepción, el aprendizaje y la resolución de problemas. El creciente poder de procesamiento de los ordenadores y la disponibilidad de grandes cantidades de datos para el entrenamiento han impulsado el rápido avance de la IA en los últimos años.

El reconocimiento de imágenes y caracteres es un campo en el que la IA ha tenido un gran impacto. La técnica de reconocimiento óptico de caracteres (OCR), se utiliza ampliamente en la digitalización de documentos, la identificación de placas de matrícula y la lectura de códigos de barras.

La IA se aplica en el reconocimiento de imágenes y caracteres mediante algoritmos de aprendizaje automático, como las redes neuronales convolucionales (CNN), que han demostrado ser muy efectivas en la clasificación de imágenes y caracteres.

La aplicación de la IA en el reconocimiento de imágenes y caracteres ha tenido un gran impacto en varias áreas, como la industria, la medicina y la seguridad.

En particular, en el ámbito de los documentos de identificación, el uso de OCR y la IA permite la automatización del proceso de identificación, reduciendo las tareas realizadas por humanos y mejorando la eficiencia.

1.2. Descripción de la nube utilizada para la construcción del sistema

En la actualidad, una de las tendencias más importantes en el ámbito de la informática es la utilización de servicios en la nube. Para la construcción del sistema propuesto en esta tesis, se ha utilizado una nube pública de proveedor Microsoft.

1.2.1. Azure

Azure es una plataforma de computación en la nube desarrollada por Microsoft, y ofrece una serie de ventajas para aquellos que utilizan sus servicios. Algunas de las ventajas clave incluyen:

- **Escalabilidad:** una de las principales ventajas de Azure es que es altamente escalable. Esto significa que es posible manejar una gran cantidad de tráfico y datos sin problemas, y se puede ajustar fácilmente para satisfacer las necesidades cambiantes del negocio.
- **Flexibilidad:** Azure ofrece una amplia gama de herramientas y servicios, lo que le permite elegir las herramientas y servicios que mejor se adapten a necesidades específicas. Asimismo, es posible elegir entre una variedad de opciones de implementación, incluyendo la nube pública, privada o híbrida.

- Seguridad: Microsoft tiene una sólida reputación en materia de seguridad y Azure no es una excepción. La plataforma cuenta con múltiples capas de seguridad para proteger los datos, incluyendo autenticación de dos factores, encriptación de datos y protección contra DDoS.
- Costo-efectividad: Azure ofrece una amplia gama de opciones de precios, lo que le permite elegir la opción que mejor se adapte al presupuesto y necesidades del negocio. Además, solo se paga por los servicios que se utilizan, lo que puede ayudar a reducir los costos generales.
- Integración: Azure es compatible con una amplia gama de herramientas y tecnologías, lo que le permite integrarse fácilmente con otras herramientas y sistemas que ya se utilizan en el negocio.

1.3. Reconocimiento óptico de caracteres (OCR)

El reconocimiento óptico de caracteres (OCR), es una técnica que se utiliza ampliamente en la digitalización de textos impresos o manuscritos, convirtiéndolos en formato digital editable, buscable y analizable.

- En el ámbito empresarial, el OCR se ha convertido en una herramienta valiosa para automatizar procesos y mejorar la eficiencia en la gestión de documentos en papel.
- La tecnología OCR se basa en algoritmos que analizan imágenes y buscan patrones de píxeles que corresponden a letras, números y otros caracteres. Una vez identificados los caracteres, el OCR los reconstruye en un formato digital que se puede procesar automáticamente.

- El OCR es utilizado en varios campos, como:
 - La digitalización de archivos históricos
 - Bibliotecas
 - Automatización de la gestión documental en empresas

- En el ámbito de la salud, el OCR se utiliza para:
 - Digitalizar historias clínicas
 - Otros documentos médicos, lo que mejora la eficiencia y la precisión en la atención médica.

- En el sector educativo, el OCR se utiliza para:
 - Digitalizar libros
 - Otros materiales educativos, lo que permite a los estudiantes acceder a una amplia variedad de recursos de aprendizaje en formato digital.

- El OCR ha permitido el desarrollo de tecnologías de accesibilidad para personas con discapacidades visuales. Por ejemplo, los lectores de pantalla utilizan el OCR para leer en voz alta el texto en las páginas web, lo que les permite acceder a la información en línea.

- En el contexto de los documentos personales de identificación, el OCR se utiliza para:
 - Digitalizar pasaportes
 - Licencias de conducir

- Tarjetas de identificación
- Otros documentos importantes

Esto no solo mejora la eficiencia en el procesamiento y la verificación de los documentos, sino que también puede reducir el tiempo y trabajo repetitivo realizado por humanos.

1.4. Documentos personales de identificación de Guatemala y sus características

En Guatemala, los documentos personales de identificación más comunes son el Documento Personal de Identificación (DPI) y La licencia de conducir. Emitidos por el Registro Nacional de las Personas (RENAP) y por el Departamento de Tránsito de la Policía Nacional Civil.

1.4.1. Documento personal de identificación (DPI)

El Documento Personal de Identificación (DPI), es emitido por el Registro Nacional de las Personas (RENAP), en Guatemala y tiene como objetivo identificar a los ciudadanos guatemaltecos en todo el país.

El DPI es esencial para realizar muchas actividades, como abrir una cuenta bancaria, solicitar un pasaporte y votar en elecciones, entre otras. Por lo tanto, es importante que los ciudadanos guatemaltecos mantengan su DPI actualizado y en buenas condiciones, ya que es su principal forma de identificación en el país.

Igualmente, el DPI cuenta con medidas de seguridad para prevenir la falsificación y el uso fraudulento del documento, como la incorporación de

características biométricas, tales como fotografías faciales. De esta forma, se garantiza la autenticidad del documento.

Los datos que se pueden encontrar en la parte frontal son:

- Código único de identificación (CUI)
- Nombres
- Apellidos
- Nacionalidad
- País de nacimiento
- Sexo
- Fecha de nacimiento
- Firma
- Fotografía
- Fecha de emisión

Figura 1. **DPI parte frontal**



Fuente: [Fotografía de José Luis Herrera Martínez]. (Guatemala, Guatemala. 2023). Colección particular. Guatemala.

El gobierno de Guatemala establece procedimientos rigurosos para emitir licencias de conducir, garantizando que solo se otorguen a personas que cumplan con los requisitos legales y cuya identidad haya sido verificada adecuadamente.

Esta licencia incluye información clave de identificación, como el nombre completo, la fecha de nacimiento, la fotografía del titular y su número de identificación personal. Dicha información se utiliza para verificar la identidad del titular, lo que convierte la licencia de conducir en un medio válido para la verificación de identidad en instituciones y empresas, como las entidades financieras.

Es importante mencionar que, aunque la verificación de identidad gracias a la licencia de conducir no es infalible, el proceso de emisión de licencias de conducir en Guatemala es muy riguroso e incluye elementos de seguridad, como marcas de agua, hologramas y tecnología de lectura magnética, lo que la convierte en un medio fiable y ampliamente aceptado para la verificación de identidad de una persona.

Los datos que se pueden encontrar en la parte frontal son:

- Nombres
- Apellidos
- Número de licencia
- Número de documento
- Firma
- Tipo de licencia
- Fecha de vencimiento

Figura 3. Licencia de conducir de Guatemala parte frontal



Fuente: [Fotografía de Victor Manuel Sagastume Ruano]. (Guatemala, Guatemala. 2023).
Colección particular. Guatemala.

Los datos que se encuentran en la parte trasera son:

- Fecha de nacimiento
- Restricciones
- Tipo de sangre
- Antigüedad
- Huella

Figura 4. **Licencia de conducir de Guatemala parte trasera**



Fuente: [Fotografía de Victor Manuel Sagastume Ruano]. (Guatemala, Guatemala. 2023).
Colección particular. Guatemala.

2. IDENTIFICACIÓN DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN

2.1. Antecedentes

El reconocimiento óptico de caracteres (OCR), es una tecnología que se ha utilizado durante décadas para convertir imágenes de texto en texto digital. El OCR se ha aplicado en diversas áreas, como en la digitalización de libros, la automatización de procesos de negocios, la creación de sistemas de ayuda a la lectura y en la identificación y clasificación de documentos.

En Guatemala, el uso del OCR se ha implementado en diversas instituciones gubernamentales y privadas para procesar y gestionar grandes volúmenes de información contenida en documentos físicos. Sin embargo, aún existen desafíos en el proceso de digitalización de los documentos personales de identificación, como la Licencia de Conducir y el DPI, debido a la complejidad de los caracteres y la gran cantidad de información que contienen.

Para abordar estos desafíos, se ha desarrollado un sistema de OCR que utiliza técnicas de inteligencia artificial para mejorar la precisión en la identificación y extracción de texto de los documentos. Este sistema de OCR basado en inteligencia artificial en la nube permite la integración de grandes cantidades de datos y el procesamiento de información en tiempo real, lo que los convierte en herramientas útiles para diversas aplicaciones.

En este contexto, el tema de tesis *Sistema de reconocimiento óptico de caracteres de los documentos personales de identificación de Guatemala,*

usando inteligencia artificial en la nube, busca desarrollar un sistema de OCR que permita la identificación y extracción precisa de los caracteres de la Licencia de Conducir y el DPI guatemaltecos, utilizando técnicas de inteligencia artificial en la nube.

El objetivo de este sistema es mejorar la eficiencia en el procesamiento de información contenida en estos documentos, lo que puede tener un impacto significativo en la gestión de información en el país.

2.2. Descripción del problema

El problema que se busca resolver es la dificultad para reconocer los datos contenidos en los documentos personales de identificación de Guatemala, específicamente en los documentos personales de identificación de Guatemala: la Licencia de Conducir y el DPI.

La Licencia de Conducir y el DPI contienen una gran cantidad de información, como el nombre completo del titular, la fecha de nacimiento, la dirección y otros datos personales. Estos documentos son utilizados en una variedad de aplicaciones, como la identificación de personas, la verificación de identidad y la gestión de información gubernamental y privada.

El problema es que, actualmente, la información contenida en estos documentos se procesa manualmente, lo que puede ser un proceso lento y propenso a errores. Igualmente, estos documentos suelen estar con diferentes fuentes, lo que dificulta la identificación y extracción precisa de los caracteres.

La solución propuesta es el desarrollo de un sistema de reconocimiento óptico de caracteres (OCR), que permita la identificación y extracción precisa de

los caracteres de la Licencia de Conducir y el DPI guatemaltecos, utilizando técnicas de inteligencia artificial en la nube.

La implementación de este sistema puede mejorar la eficiencia y precisión en el procesamiento de información contenida en estos documentos, lo que puede tener un impacto significativo en la gestión de información en el país.

2.3. Mercado objetivo

El mercado objetivo son todas las instituciones públicas y privadas que requieren procesar grandes cantidades de información contenida en los documentos personales de identificación de Guatemala, específicamente la Licencia de Conducir y el DPI.

Entre las instituciones públicas que podrían beneficiarse de este sistema se encuentran el Registro Nacional de las Personas (RENAP), el Departamento de Tránsito de la Policía Nacional Civil, el Ministerio de Gobernación, el Ministerio de Relaciones Exteriores, entre otros. Estas instituciones procesan grandes cantidades de información personal contenida en los documentos de identificación, y la implementación de un sistema de OCR basado en inteligencia artificial en la nube podría mejorar significativamente su eficiencia y precisión en el procesamiento de información.

Asimismo, las instituciones privadas que requieren procesar información personal contenida en los documentos de identificación, como empresas de telecomunicaciones, bancos, compañías de seguros, entre otras, también podrían ser un mercado objetivo para este sistema de reconocimiento óptico de caracteres.

3. DOCUMENTO DE ESPECIFICACIONES, REQUERIMIENTOS Y CRITERIOS DE ACEPTACIÓN

3.1. Requerimientos

Para el desarrollo del sistema propuesto en esta tesis, los requerimientos se clasifican como:

3.1.1. Requerimientos funcionales

- Captura de imágenes: el sistema debe ser capaz de capturar imágenes digitales claras de los documentos de identificación, como la Licencia de Conducir y el DPI.
- Procesamiento de imágenes: el sistema debe procesar las imágenes capturadas para ajustar el brillo, el contraste y otros aspectos para mejorar la calidad de las imágenes.
- Reconocimiento óptico de caracteres: el sistema debe ser capaz de reconocer y extraer los caracteres de la imagen y convertirlos en texto legible.
- Integración con otros sistemas: el sistema debe ser compatible con otros sistemas y plataformas, para poder compartir la información de manera eficiente con otras instituciones.

- Interfaz de usuario: el sistema debe tener una interfaz de usuario intuitiva y fácil de usar para poder probar el API del sistema desarrollado.

3.1.2. Requerimientos no funcionales

- Rendimiento: el sistema debe ser capaz de procesar imágenes y reconocer caracteres de manera rápida y eficiente, para minimizar los tiempos de espera para los usuarios.
- Escalabilidad: el sistema debe ser escalable para manejar grandes volúmenes de imágenes y usuarios simultáneos, sin afectar el rendimiento.
- Disponibilidad: el sistema debe estar disponible en todo momento, para garantizar el acceso a los datos de identificación en cualquier momento que sea necesario.
- Tolerancia a fallos: el sistema debe ser capaz de recuperarse de los errores y fallas, para minimizar el impacto en la disponibilidad y el rendimiento del sistema.
- Usabilidad: la interfaz de usuario debe ser intuitiva y fácil de usar, para que los usuarios puedan interactuar con el sistema de manera efectiva y eficiente.

3.2. Alcances

- Captura de imágenes: el sistema permitirá a los usuarios capturar imágenes de los documentos de identificación, como el DPI y la licencia de conducir, utilizando una cámara o escáner.

- **Procesamiento de imágenes:** el sistema procesará las imágenes capturadas para extraer la información de texto utilizando técnicas de procesamiento de imágenes y reconocimiento óptico de caracteres.
- **Almacenamiento de la información:** el sistema almacenará la información de identificación en una base de datos segura y protegida.
- **Integración con otros sistemas:** el sistema será compatible con otros sistemas y servicios, como bases de datos gubernamentales, para facilitar el intercambio de información y mejorar la eficiencia.
- **Acceso y consulta de la información:** los usuarios autorizados podrán acceder y consultar la información de identificación almacenada en el sistema, de manera segura y confiable.
- **Seguridad y privacidad:** el sistema implementará medidas de seguridad y privacidad para proteger la información de identificación de los usuarios.

3.3. Limitantes

- **Calidad de las imágenes:** el sistema puede no ser capaz de procesar imágenes de baja calidad, lo que puede reducir la precisión del reconocimiento de caracteres.
- **Variaciones en el formato de los documentos:** si los documentos tienen variaciones en su formato o diseño, el sistema puede tener dificultades para reconocer el texto correctamente.

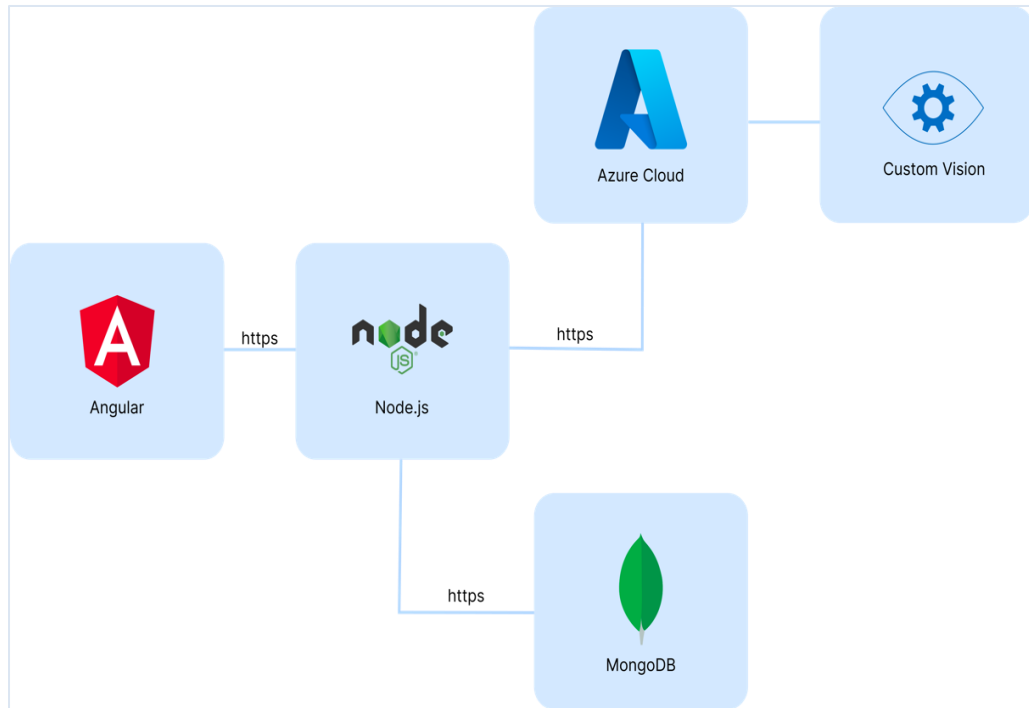
- Requerimientos de ancho de banda y procesamiento: el procesamiento de imágenes y el reconocimiento óptico de caracteres pueden requerir una gran cantidad de ancho de banda y recursos de procesamiento, lo que puede limitar la eficiencia del sistema.
- Acceso a los datos de identificación: el sistema solo puede acceder a la información de identificación almacenada en los documentos de identificación, y no a bases de datos oficiales gubernamentales que contienen información adicional.

3.4. Arquitectura de la solución

La solución se ha diseñado utilizando una arquitectura cliente-servidor. El cliente se ha implementado utilizando el framework de Javascript Angular, mientras que en el servidor se ha utilizado el entorno de ejecución de Javascript Node.js junto con el framework Express.js. También, se ha establecido una conexión con la API de Azure para aprovechar el servicio de Custom Vision y se ha integrado una base de datos no relacional MongoDB.

A continuación, se presenta el diagrama de la solución propuesta.

Figura 5. **Arquitectura de la solución**



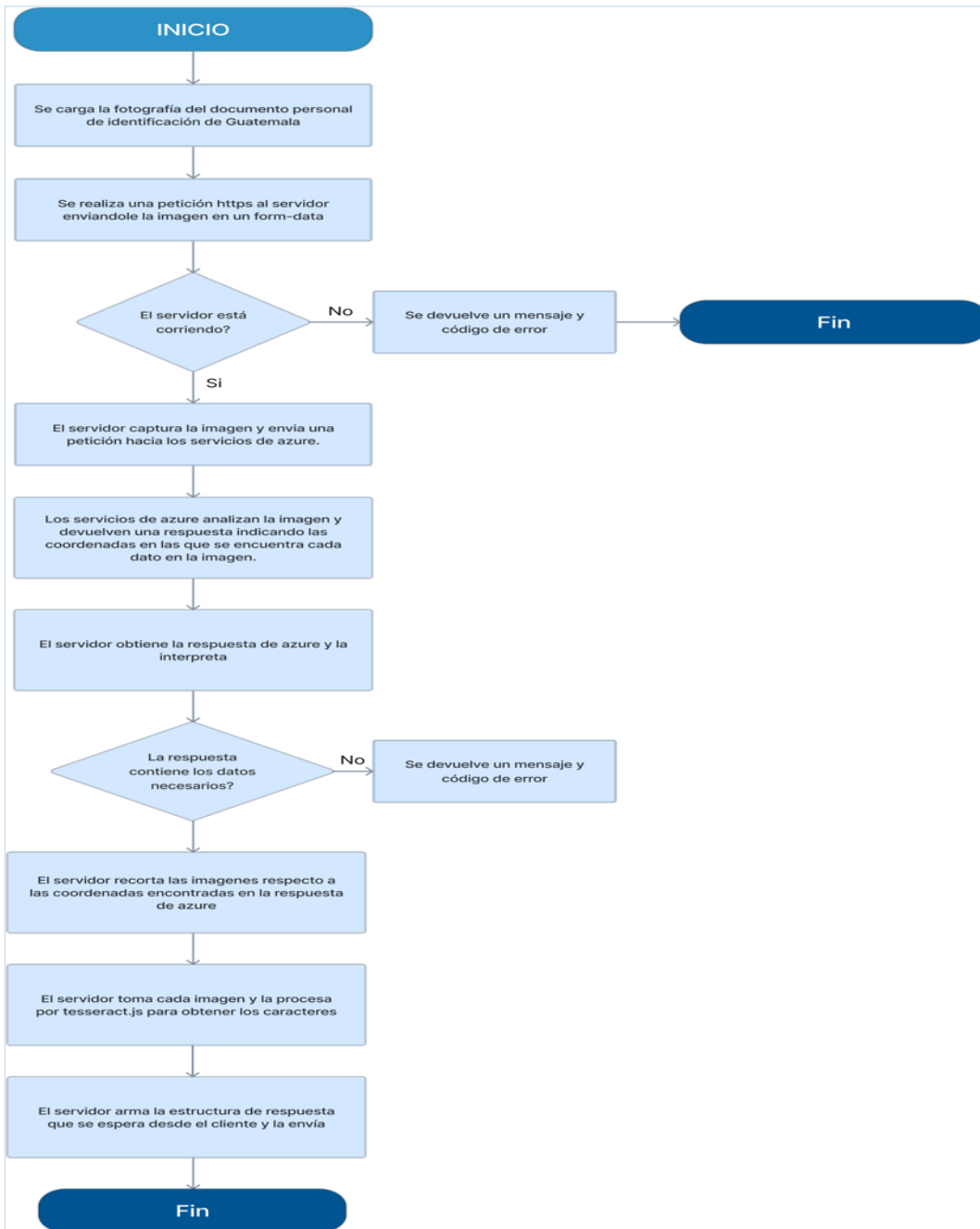
Fuente: elaboración propia, realizado con Figma.

3.5. Diagrama de flujo de la solución

Un diagrama de flujo es una herramienta de representación gráfica que se utiliza para visualizar el flujo de información, procesos, decisiones o cualquier otra actividad que se realice en un sistema.

Este tipo de diagrama utiliza símbolos y flechas para representar los diferentes pasos del proceso y las relaciones entre ellos.

Figura 6. Diagrama de flujo de la solución



Fuente: elaboración propia, realizado con Figma.

4. DISEÑO DE LA APLICACIÓN

4.1. Descripción del software

Se propone el diseño de un sistema basado en una arquitectura cliente-servidor. En el lado del cliente, se pueden probar las funcionalidades de OCR para los documentos de identificación personal de Guatemala, mientras que el servidor se encarga de la lógica y el consumo de los servicios en la nube de Azure para procesar la solicitud de OCR.

El proceso de la aplicación es el siguiente: se carga una imagen en cualquier formato desde el lado del cliente y se envía al servidor mediante una solicitud HTTPS utilizando el método POST. El servidor lee la imagen y la envía a la API de Azure, Custom Vision, para que pueda procesar la imagen y reconocer las regiones relevantes que contienen la información que se desea extraer mediante OCR. La API de Custom Vision devuelve una respuesta en formato JSON que contiene las coordenadas de las regiones donde se encuentra la información.

Luego, el servidor interpreta la respuesta y recorta la imagen en las regiones correspondientes utilizando las coordenadas proporcionadas. Cada imagen recortada se procesa individualmente para convertirla a escala de grises y reducir el ruido de la imagen. Después, se utiliza una librería llamada Tesseract.js para aplicar OCR y extraer el texto de las imágenes. La información extraída se almacena temporalmente en la memoria del servidor.

Posteriormente se almacena la información en MongoDB para poder tener un registro de todos los documentos que se han reconocido y que estén disponibles en cualquier momento que sea deseado que sea consultado.

Finalmente, el servidor devuelve la información extraída en formato JSON al cliente, y el cliente interpreta la respuesta y la renderiza para que sea visualizada.

4.2. Diseño de la aplicación web

El diseño de la aplicación web es un elemento clave en el desarrollo del sistema propuesto en esta tesis. El objetivo del diseño es proporcionar una interfaz de usuario intuitiva y fácil de usar para los usuarios finales.

4.2.1. Vistas

Las vistas del frontend son una parte fundamental del sistema propuesto en esta tesis. Estas vistas proporcionan la interfaz de usuario que los usuarios utilizarán para interactuar con el sistema.

4.2.1.1. Página Home

Esta es la primera pantalla que se carga al momento de ingresar a la aplicación web y se puede seleccionar que servicio se quiere utilizar, ya sea OCR DPI u OCR Licencia de conducir.

Figura 7. **Página Home de la aplicación web**

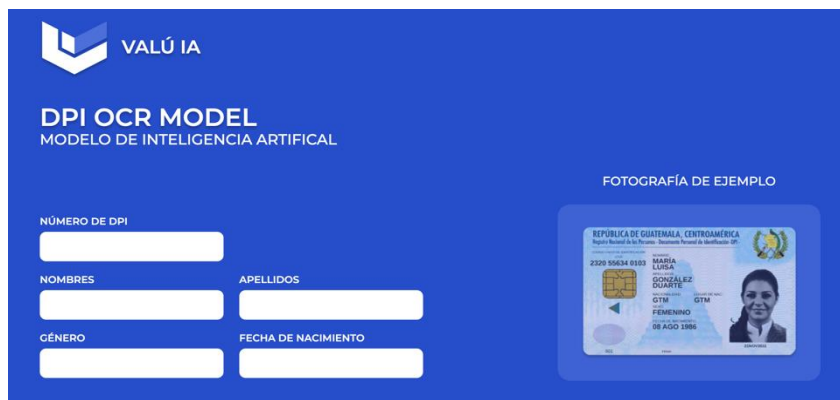


Fuente: elaboración propia, realizado con captura de pantalla.

4.2.1.2. **Página OCR DPI**

Esta página en específico permite utilizar el servicio de OCR para el DPI, se puede adjuntar la imagen del documento y posteriormente despliega la información del documento.

Figura 8. **Página OCR DPI de la aplicación web**

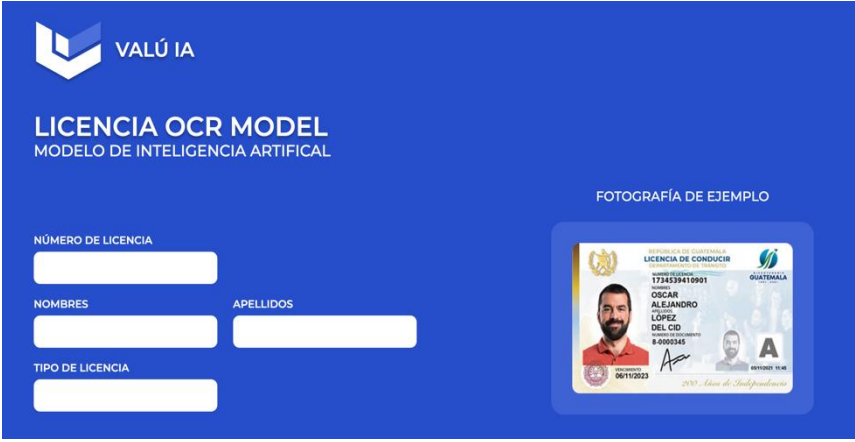


Fuente: elaboración propia, realizado con captura de pantalla.

4.2.1.3. Página OCR Licencia de conducir

Esta página en específico permite utilizar el servicio de OCR para la licencia de conducir, se puede adjuntar la imagen del documento y posteriormente despliega la información del documento.

Figura 9. **Página OCR Licencia de la aplicación web**



Fuente: elaboración propia, realizado con captura de pantalla.

4.3. Diseño del API

El diseño del API es un aspecto crítico en el desarrollo del sistema propuesto en esta tesis ya que es la encargada de proveer la información necesaria para poder desplegar en el sistema.

4.3.1. Endpoints

La API cuenta con tres endpoints principales para interactuar con los datos de los documentos de identificación almacenados. Para acceder y consumir los datos de los endpoints, es necesario enviar un JWT para autenticarse de manera segura y efectiva.

4.3.1.1. Datos del DPI

El propósito de este endpoint es devolver los datos de un documento de identificación personal (DPI), específico mediante el uso del número de DPI correspondiente. La fotografía del DPI se devuelve en formato base64. Para acceder a este endpoint, es necesario enviar una solicitud HTTP POST.

En seguida se pueden observar ejemplos de los datos de entrada y de salida del endpoint.

Figura 10. Ejemplo de request para el endpoint datos del dpi

```
{  
  "dpi": "300876530102"  
}
```

Fuente: elaboración propia, realizado con captura de pantalla.

Figura 11. **Ejemplo de response para el endpoint datos del dpi**

```
{
  "numero": "300876530102",
  "nombres": "Jose Luis",
  "apellidos": "Herrera Martinez",
  "fechaNacimiento": "27/06/1999",
  "sexo": "M",
  "foto": "<BASE_64_IMAGE>"
}
```

Fuente: elaboración propia, realizado con captura de pantalla.

4.3.1.2. **Datos de la licencia de conducir**

Este endpoint se encarga de devolver los datos de un DPI específico mediante el número de DPI proporcionado. La fotografía se devuelve en formato base64. Para acceder a este endpoint, es necesario utilizar el verbo HTTP POST.

A continuación, se pueden observar ejemplos de los datos de entrada y de salida del endpoint.

Figura 12. **Ejemplo de request para el endpoint datos de la licencia de conducir**

```
{
  "licencia": "123456789"
}
```

Fuente: elaboración propia, realizado con captura de pantalla

Figura 13. **Ejemplo de response para el endpoint datos de la licencia de conducir**

```
{
  .... "numero": "123456789",
  .... "tipo": "C",
  .... "nombres": "Jose Luis",
  .... "apellidos": "Herrera Martinez",
  .... "fechaNacimiento": "27/06/1999",
  .... "sexo": "M",
  .... "foto": "<BASE_64_IMAGE>"
}
```

Fuente: elaboración propia, realizado con captura de pantalla.

4.3.1.3. **Obtener todos los registros**

El objetivo de este endpoint es listar todos los documentos que han sido procesados por el sistema OCR. Para acceder a este método, es necesario utilizar el verbo HTTP GET.

A continuación, se pueden observar ejemplos de los datos de entrada y de salida del endpoint.

Figura 14. **Ejemplo de response para el endpoint obtener todos los datos**

```
{
  "analizados": [
    {
      "documento": "Licencia",
      "numero": "123456789",
      "tipo": "C",
      "nombres": "Jose Luis",
      "apellidos": "Herrera Martinez",
      "fechaNacimiento": "27/06/1999",
      "sexo": "M",
      "foto": "<BASE_64_IMAGE>"
    },
    {
      "documento": "DPI",
      "numero": "300876389701",
      "nombres": "Jose Luis",
      "apellidos": "Herrera Martinez",
      "fechaNacimiento": "27/06/1999",
      "sexo": "M",
      "foto": "<BASE_64_IMAGE>"
    }
  ]
}
```

Fuente: elaboración propia, realizado con captura de pantalla.

4.4. Datos de entrenamiento

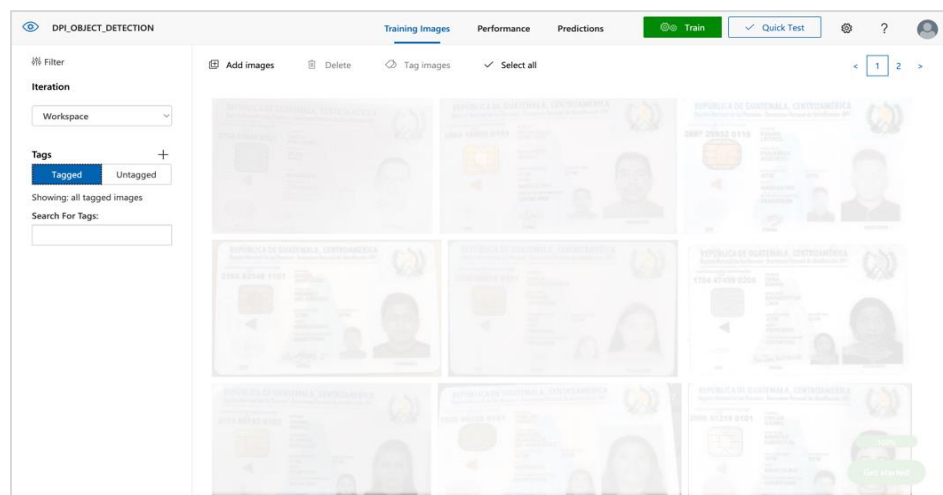
El modelo de reconocimiento de imágenes utilizado en este software se entrenó utilizando un conjunto de imágenes representativas del DPI y la licencia de conducir. Para el modelo de DPI, se utilizaron un total de 100 imágenes para entrenar el modelo de inteligencia artificial. Estas imágenes se seleccionaron cuidadosamente para incluir una variedad de ángulos, iluminación y características para mejorar la precisión del modelo.

Por otro lado, para el modelo de reconocimiento de licencia de conducir, se utilizaron un total de 80 imágenes para entrenar al modelo de IA. De manera similar al modelo de DPI, se seleccionaron cuidadosamente estas imágenes para asegurar que representen una variedad de características y detalles de la licencia de conducir.

Es importante tener en cuenta que la precisión del modelo puede depender en gran medida de la calidad y cantidad de las imágenes utilizadas para entrenarlo. Al utilizar un número adecuado de imágenes, se puede mejorar la precisión del modelo y reducir el riesgo de errores al identificar los documentos.

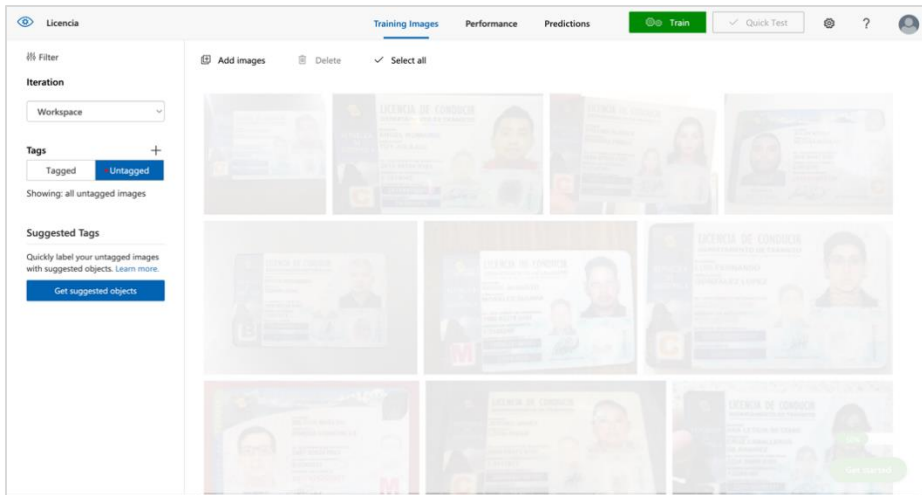
Se presentan algunos ejemplos de las imágenes utilizadas para entrenar los modelos que reconocen las regiones que contienen la información de los documentos personales de identificación.

Figura 15. **Datos de entremiento DPI Custom Vision Azure**



Fuente: elaboración propia, realizado con captura de pantalla.

Figura 16. **Datos de entrenamiento Licencia Custom Vision Azure**



Fuente: elaboración propia, realizado con captura de pantalla.

4.5. **Herramientas y tecnologías utilizadas en el desarrollo del sistema**

El desarrollo del sistema propuesto en esta tesis ha requerido el uso de diversas herramientas y tecnologías para su implementación. En esta sección se describirán las principales herramientas y tecnologías utilizadas en el desarrollo del sistema.

4.5.1. **Detección de objetos de Azure**

Azure Object Detection es un servicio de Microsoft Azure que permite detectar objetos en imágenes y videos utilizando algoritmos de aprendizaje automático y redes neuronales convolucionales. Este servicio es parte de Azure Cognitive Services, una plataforma de inteligencia artificial que permite a los desarrolladores integrar características de inteligencia artificial en sus aplicaciones.

Con Azure Object Detection, los desarrolladores pueden agregar capacidades de detección de objetos en sus aplicaciones sin necesidad de construir desde cero modelos de aprendizaje automático. En cambio, pueden enviar imágenes y videos por una API, y recibir resultados de detección de objetos.

El servicio de Azure Object Detection admite la detección de varios objetos en una sola imagen o video y puede identificar la ubicación, tamaño y forma de cada objeto detectado. También puede etiquetar objetos con información adicional, como su tipo o categoría, lo que puede ser útil para tareas de clasificación.

Azure Object Detection se puede usar en una variedad de escenarios, como en la detección de rostros en imágenes, detección de vehículos en videos de tráfico, análisis de imágenes médicas y más. La API de Azure Object Detection se puede integrar fácilmente en aplicaciones web y móviles mediante la utilización de kits de desarrollo de software (SDK), y lenguajes de programación como .NET, Java, Python y JavaScript.

4.5.2. Angular

Angular es un popular framework de JavaScript desarrollado y mantenido por Google. Permite a los desarrolladores construir aplicaciones web dinámicas y de una sola página (SPA), de manera rápida y eficiente. Angular está diseñado para trabajar con el patrón de arquitectura Model-View-Controller (MVC), y Model-View-ViewModel (MVVM), lo que facilita la organización y el mantenimiento del código.

Angular está basado en TypeScript, un lenguaje de programación que es una extensión de JavaScript y que proporciona características adicionales, como la verificación de tipos estáticos, lo que hace que el código sea más seguro y fácil de mantener. Además, Angular incluye un conjunto de bibliotecas y herramientas que permiten la creación de componentes personalizados, servicios, directivas y mucho más.

Entre las principales características de Angular se encuentran la inyección de dependencias, que permite una mejor modularidad del código; la vinculación de datos, que sincroniza automáticamente los cambios entre los datos del modelo y la vista de la aplicación; y la animación, que permite agregar efectos visuales a la aplicación.

Angular ha sido utilizado por empresas y organizaciones de todo el mundo para construir aplicaciones web de alta calidad. Su popularidad se debe en parte a su amplia documentación y a la comunidad de desarrolladores que lo respalda, lo que facilita el aprendizaje y la resolución de problemas.

4.5.3. Node JS

Node.js es un entorno de tiempo de ejecución de JavaScript que se utiliza para ejecutar código del lado del servidor. Node.js se basa en el motor de JavaScript V8 de Google, que es el mismo motor utilizado por el navegador web Google Chrome. Sin embargo, Node.js no se utiliza para ejecutar JavaScript en un navegador, sino para ejecutar JavaScript en un servidor.

Node.js es conocido por ser rápido y escalable, y es especialmente adecuado para aplicaciones en tiempo real y de alta carga, como aplicaciones de chat, juegos en línea, transmisiones en vivo y aplicaciones de comercio

electrónico. De igual manera, Node.js es muy popular entre los desarrolladores web debido a su facilidad de uso y su amplia comunidad de desarrolladores y herramientas.

Una de las principales características de Node.js es su capacidad para manejar múltiples conexiones simultáneas, lo que lo hace ideal para aplicaciones en tiempo real y basadas en eventos. También es capaz de manejar la entrada y salida de manera asíncrona, lo que significa que puede realizar varias tareas a la vez sin bloquear el hilo de ejecución.

Otra característica importante de Node.js es su sistema de módulos, que permite a los desarrolladores reutilizar el código y crear aplicaciones escalables y fáciles de mantener. Los módulos de Node.js se pueden instalar y administrar fácilmente mediante npm, el gestor de paquetes de Node.js.

4.5.4. Express JS

Express.js es un framework de JavaScript utilizado para desarrollar aplicaciones web en Node.js. Es una de las herramientas más populares y ampliamente utilizadas para desarrollar aplicaciones web en Node.js debido a su facilidad de uso y su flexibilidad.

Express.js proporciona una amplia variedad de funciones y características, como el enrutamiento, la gestión de solicitudes HTTP, el procesamiento de middleware, y la integración con motores de plantillas, que facilitan el desarrollo de aplicaciones web en Node.js. Asimismo, Express.js es altamente personalizable y se puede integrar con otros módulos y bibliotecas de Node.js, lo que permite a los desarrolladores crear aplicaciones web altamente especializadas y personalizadas.

Una de las características más importantes de Express.js es su capacidad para manejar solicitudes HTTP y enrutarlas a la función correspondiente en función de la URL de la solicitud. Esto significa que los desarrolladores pueden crear una API RESTful para su aplicación web y manejar solicitudes GET, POST, PUT y DELETE, entre otros métodos HTTP.

Otra característica importante de Express.js es su capacidad para procesar middleware, que son funciones que se ejecutan antes de que se maneje una solicitud. Esto significa que los desarrolladores pueden agregar funciones personalizadas a su aplicación, como la autenticación, el registro de solicitudes, la compresión de archivos y la gestión de errores.

Express.js también es altamente personalizable, lo que significa que los desarrolladores pueden agregar y quitar funciones según sea necesario para satisfacer las necesidades específicas de su aplicación web. De igual forma, hay una gran cantidad de bibliotecas y módulos disponibles en la comunidad de Node.js que se pueden integrar con Express.js, lo que permite a los desarrolladores agregar nuevas funciones y características a su aplicación de manera rápida y sencilla.

4.5.5. Tesseractjs

Tesseract.js es una biblioteca de reconocimiento óptico de caracteres (OCR), basada en JavaScript que permite a los desarrolladores de aplicaciones web y móviles reconocer y extraer texto de imágenes y PDFs. Tesseract.js es una implementación de Tesseract, una herramienta de OCR de código abierto que ha sido desarrollada por Google.

Una de las principales ventajas de Tesseract.js es que es compatible con varios formatos de imagen y PDFs, incluyendo JPEG, PNG y TIFF, lo que permite a los desarrolladores trabajar con una amplia variedad de tipos de archivos. De igual manera, Tesseract.js es fácil de integrar en aplicaciones web y móviles, ya que se ejecuta en el navegador web o en el servidor, y se puede utilizar en conjunto con otros lenguajes de programación y tecnologías, como Node.js y React.

Otra ventaja de Tesseract.js es que tiene una gran precisión en el reconocimiento de caracteres, lo que significa que puede reconocer y extraer texto con una alta tasa de éxito. Esto lo hace ideal para aplicaciones que requieren el reconocimiento automático de texto, como la transcripción de documentos o la captura de datos de formularios.

Tesseract.js también ofrece una amplia variedad de configuraciones y opciones que permiten a los desarrolladores personalizar el proceso de reconocimiento de caracteres según sus necesidades específicas. Por ejemplo, los desarrolladores pueden ajustar la configuración para mejorar la precisión del reconocimiento de caracteres, configurar el idioma para reconocer texto en idiomas específicos, y definir las regiones de la imagen donde se debe buscar el texto.

4.5.6. MongoDB

MongoDB es un sistema de base de datos NoSQL que se caracteriza por ser un sistema de base de datos de documentos. Esto significa que, en lugar de guardar la información en tablas como en las bases de datos relacionales, MongoDB guarda la información en documentos en formato BSON (Binary

JSON), que es un formato de datos binario que permite la serialización y deserialización de documentos en diferentes lenguajes de programación.

Una de las principales características de MongoDB es su capacidad para escalar horizontalmente. Esto significa que es posible agregar más servidores para distribuir la carga de trabajo y aumentar el rendimiento del sistema sin tener que cambiar la estructura de la base de datos. También, MongoDB cuenta con un lenguaje de consulta llamado MongoDB Query Language (MQL), que permite realizar consultas complejas y flexibles en la base de datos.

Otra característica importante de MongoDB es su flexibilidad en cuanto al esquema de la base de datos. A diferencia de las bases de datos relacionales que tienen un esquema rígido, MongoDB permite cambiar el esquema de la base de datos a medida que las necesidades del sistema cambian. Esto hace que sea fácil agregar nuevos campos o modificar la estructura de los documentos sin tener que hacer cambios en toda la base de datos.

MongoDB es ampliamente utilizado en aplicaciones web y móviles que manejan grandes cantidades de datos y necesitan una base de datos escalable y flexible. También es popular en aplicaciones de Internet de las cosas (IoT), y en análisis de datos en tiempo real debido a su capacidad para manejar grandes cantidades de datos y su capacidad de consulta flexible.

4.5.7. Visual Studio Code

Visual Studio Code, comúnmente abreviado como VS Code, es un editor de código fuente desarrollado por Microsoft. Es un software libre y de código abierto que se ejecuta en múltiples plataformas, incluyendo Windows, Linux y macOS. VS Code es una herramienta popular y ampliamente utilizada por los

desarrolladores de software debido a su facilidad de uso y su gran cantidad de características y extensiones.

VS Code es un editor de texto avanzado que incluye características como poder resaltar la sintaxis, la sugerencia de código, la depuración integrada, la gestión de repositorios y la integración con herramientas externas. A su vez, ofrece una amplia variedad de extensiones desarrolladas por la comunidad, que permiten a los usuarios personalizar el editor y agregar nuevas funcionalidades.

Una de las características más populares de VS Code es su integración con Git, el sistema de control de versiones más utilizado en el mundo del desarrollo de software. Los desarrolladores pueden clonar, hacer cambios y crear nuevas ramas en los repositorios de Git directamente desde el editor, lo que facilita el flujo de trabajo del desarrollo de software.

A su vez, VS Code es altamente personalizable y configurable, lo que significa que los usuarios pueden agregar y quitar características y extensiones según sea necesario para satisfacer sus necesidades específicas. También incluye una gran cantidad de atajos de teclado que permiten a los usuarios trabajar de manera más eficiente y productiva.

Otra característica importante de VS Code es su capacidad para depurar código en múltiples lenguajes de programación, lo que facilita la identificación y resolución de errores en el código. Además, VS Code es compatible con una amplia variedad de lenguajes de programación, incluyendo JavaScript, Python, C++, y muchos otros, lo que lo hace una excelente opción para desarrolladores que trabajan con múltiples lenguajes.

5. EVALUACIÓN DEL SISTEMA

5.1. Análisis de resultados obtenidos del modelo para DPI

La precisión en IA se refiere a la capacidad del modelo para clasificar correctamente las instancias que se le presentan. Es decir, mide la proporción de predicciones que son verdaderamente positivas (verdaderos positivos), entre el total de predicciones. Una precisión del 100 % indica que el modelo clasifica todas las instancias correctamente.

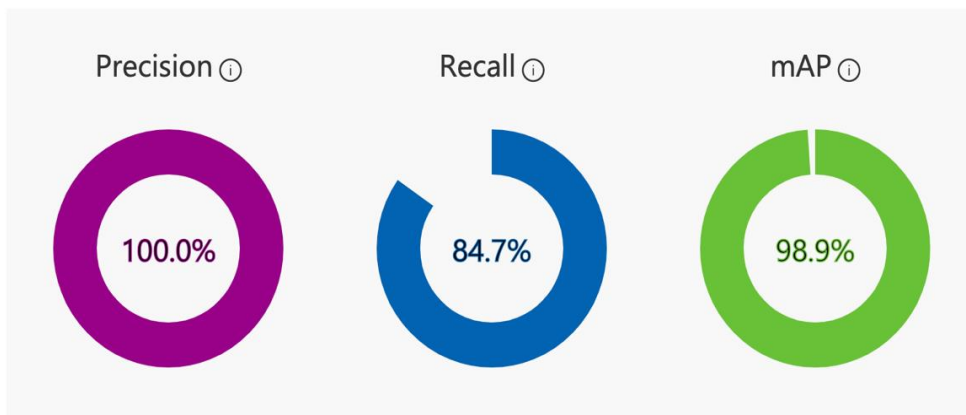
El recall, por otro lado, se refiere a la capacidad del modelo para encontrar todas las instancias positivas en los datos. Es decir, mide la proporción de instancias positivas que el modelo clasifica correctamente entre todas las instancias positivas en los datos. Un recall del 84.7 % indica que el modelo es capaz de encontrar correctamente el 84.7 % de todas las instancias positivas en los datos.

El mAP (mean average precision), es una métrica que combina tanto la precisión como el recall para evaluar el rendimiento general de un modelo de detección de objetos. El mAP calcula el área debajo de la curva de precisión-recall (PR), y proporciona una medida agregada del rendimiento del modelo. Un mAP del 98.9 % indica que el modelo tiene un rendimiento muy bueno en la detección de objetos en las imágenes.

En resumen, los resultados obtenidos indican que el modelo entrenado con la plataforma de Azure Custom Vision tiene una precisión del 100 %, es capaz de encontrar correctamente el 84.7 % de todas las instancias positivas en

los datos y tiene un rendimiento general muy bueno en la detección de objetos en las imágenes, con un mAP del 98.9 %. Estos resultados muestran la eficacia del modelo y su capacidad para clasificar y detectar objetos en imágenes de documentos de identificación personal de Guatemala.

Figura 17. **Resultados obtenidos Custom Vision Azure DPI**



Fuente: elaboración propia, realizado con Custom Vision de Azure.

5.2. **Análisis de resultados obtenidos del modelo para Licencia de conducir**

Los resultados obtenidos indican que el modelo entrenado con la plataforma de Azure Custom Vision tiene una precisión del 85.7 %, es capaz de encontrar correctamente el 100 % de todas las instancias positivas en los datos y tiene un rendimiento general muy bueno en la detección de objetos en las imágenes, con un mAP del 100 %. Estos resultados muestran la eficacia del modelo y su capacidad para clasificar y detectar objetos en imágenes de documentos de identificación personal de Guatemala.

Figura 18. **Resultados obtenidos Custom Vision Azure Licencia de conducir**



Fuente: elaboración propia, realizado con Custom Vision de Azure.

CONCLUSIONES

1. Implementar un sistema de reconocimiento óptico de caracteres (OCR), en la nube mejora la precisión y eficiencia del proceso de conversión de imágenes a texto digital, gracias al uso de la inteligencia artificial (IA), y la capacidad de procesamiento en tiempo real.
2. La implementación de sistemas de reconocimiento óptico de caracteres (OCR), brinda una mayor rapidez en la lectura y procesamiento de documentos de identificación personal en comparación con el proceso manual realizado por un humano.
3. Para un sistema OCR basado en la nube que convierte los documentos de identificación personal de Guatemala en texto digital, fue necesario utilizar 100 imágenes del DPI y 80 imágenes de la licencia de conducir para realizar 3 entrenamientos, para alcanzar la precisión deseada mayor o igual al 85 %.

RECOMENDACIONES

1. Seleccionar imágenes de alta calidad y que sean representativas y variadas para que sean capaces de cubrir la mayor cantidad de escenarios posibles para poder obtener resultados más precisos.
2. Realizar pruebas con diversas variantes de formatos y tamaños de las imágenes, ya que estos pueden impactar en los resultados del modelo y afectar su precisión.
3. Tomar en cuenta al momento de realizar el entrenamiento del modelo, que es fundamental evitar el sobreajuste para garantizar resultados precisos. Es importante tener en cuenta que optimizar el modelo en exceso puede ser perjudicial, por lo tanto, es necesario aplicar técnicas adecuadas de regularización para obtener un modelo generalizado y evitar el sobreajuste.
4. Emplear el presente trabajo como base para poder implementar el sistema de OCR para diferentes enfoques que puedan aportar a Guatemala porque estos avances tienen el potencial de mejorar significativamente otros procesos de lectura de documentos en diferentes ámbitos, como el sector financiero, legal y gubernamental, lo que se traduciría en una mayor eficiencia y productividad en estas áreas.

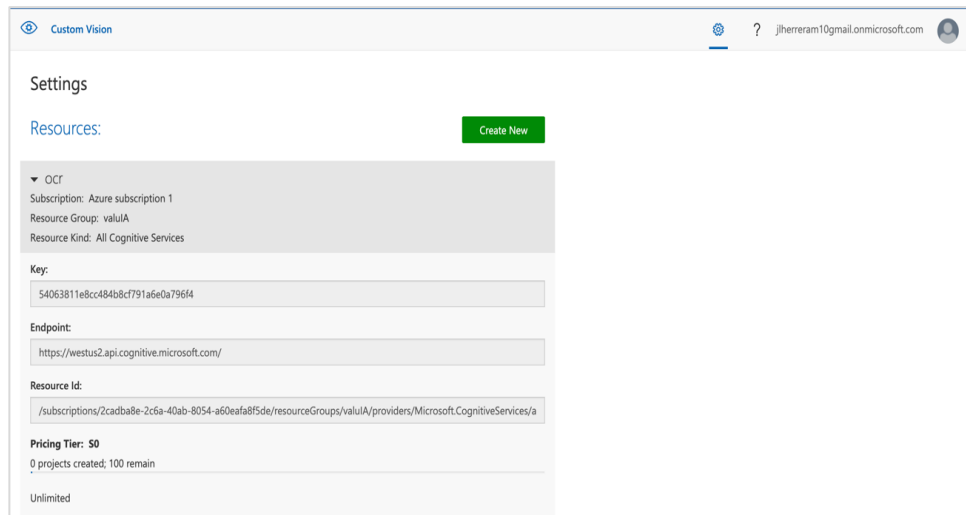
REFERENCIAS

1. Google (2010). *Angular*. Estados Unidos: Autor.
2. Microsoft (2023). *Custom Vision*. Estados Unidos: Autor.
3. MongoDB, Inc. (2023). *MongoDB*. Estados Unidos: Autor.
4. Naptha (2023). *Tesseractjs*. Estados Unidos: Autor.
5. Nodejs ORG (2023). *Nodejs*. Estados Unidos: Autor.
6. Parzibyte, A. (18 de febrero, 2023). Reconocimiento óptico de caracteres con Tesseract OCR [Mensaje en un blog]. Recuperado de <https://parzibyte.me/blog/2019/05/25/reconocimiento-optico-caracteres-tesseract-ocr>
7. StrongLoop, Inc. (2023). *Express.js*. Estados Unidos: Autor.

APÉNDICES

Para la utilización de la API de Object detection de Azure, se accede a la nube de Azure, luego se ingresa al portal de custom visión y en la sección de configuración se tienen las credenciales para poder utilizar el API.

Apéndice 1. Utilización de la API de Object detection de Azure



Fuente: elaboración propia, realizado con captura de pantalla.

Luego se declara en el archivo todas las keys que se van a utilizar para poder consumir el api de custom vision de Azure.

Apéndice 2. Credenciales para el consumo de custom vision de Azure

```
const DPI_VALUES_URL = 'https://southcentralus.api.cognitive.microsoft.com/customvision/v3.0/Prediction/94674ec5-28e9-422b-8985-47856f0ea6fd/detect/iterations/dpi_od';
const DPI_VALUES_PREDICTION_KEY = 'e3428b4c19574e37b7439da1d338b6f1';
const DPI_VALUES_RESOURCE_ID = 'dpi_od_v1';
const DPI_VALUES_PROJECT_ID = '94674ec5-28e9-422b-8985-47856f0ea6fd';
```

Fuente: elaboración propia, realizado con captura de pantalla.

Luego se importan las dependencias de npm que son necesarias para conectarse con los servicios de Azure desde node.js, también se crea el objeto predictor que es el encargado de hacer la llamada al api de Azure.

Apéndice 3. Importación de dependencias de Azure en nodejs

```
const values = require('../constants/values');
const PredictionApi = require("@azure/cognitiveservices-customvision-prediction");
const msRest = require("@azure/ms-rest-js");

const predictorCredentials = new msRest.ApiKeyCredentials({ inHeader: { "Prediction-key": values.DPI_VALUES_PREDICTION_KEY } });
const predictor = new PredictionApi.PredictionAPIClient(predictorCredentials, values.DPI_VALUES_URL);

module.exports = predictor;
```

Fuente: elaboración propia, realizado con captura de pantalla.

Posteriormente se construye un servicio (clase), que contenga el método que hará la llamada al api de Azure, el cual recibe como parámetro las credenciales que anteriormente se declararon en el código.

Apéndice 4. Servicio para hacer la llamada al API de Azure

```
const predictor = require('../config/dpi_values_prediction');
const values = require('../constants/values');

class DPIService {
  constructor() {}

  async predict(file, dimensiones) {
    const results = await predictor.detectImage(values.SMART_CROP_PROJECT_ID, values.SMART_CROP_RESOURCE_ID, file);

    const dpiValues = {};

    results.predictions.forEach(element => {
      if(!dpiValues[element.tagName]) {
        dpiValues[element.tagName] = {
          name: element.tagName,
          probability: element.probability * 100,
          box: element.boundingBox
        };
      }
      else if (dpiValues[element.tagName].probability < element.probability) {
        dpiValues[element.tagName] = {
          name: element.tagName,
          probability: element.probability * 100,
          box: element.boundingBox
        };
      }
    });

    return dpiValues;
  }
}

module.exports = DPIService;
```

Fuente: elaboración propia, realizado con captura de pantalla.

