

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



Presentado por

BACH. WRITHNEY XIMENA LAZO MARTÍNEZ

SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE
CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE
BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y
VISIÓN COMPUTACIONAL

Asesor:

M.SC. CARLOS YINMEL CASTRO
BULEJE

TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERÍA DE SISTEMAS

ANDAHUAYLAS – APURÍMAC – PERÚ

2023



APROBACIÓN DEL ASESOR

Quién suscribe:

M.Sc. Carlos Yinmel Castro Buleje con título profesional de Ingeniero de Sistemas,
por la presente:

CERTIFICA,

Que, el bachiller en Ingeniería de sistemas, **WRITHNEY XIMENA LAZO MARTÍNEZ** ha culminado satisfactoriamente el informe final de la tesis intitulada: **“SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL”** para optar el Título Profesional de Ingeniera de Sistemas.

Andahuaylas, 27 de noviembre de 2023.

M.Sc. Carlos Yinmel Castro Buleje
Asesor

Bach. Writhney Ximena Lazo Martinez
Tesista



FACULTAD DE INGENIERIA

ACTA DE SUSTENTACIÓN DE TESIS

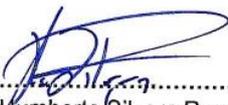
En la Av. José María Arguedas del Local Académico SL01 (Ccoyahuacho) en el auditorio de la Escuela Profesional de Ingeniería de Sistemas de la Universidad Nacional José María Arguedas ubicado en el distrito de San Jerónimo de la Provincia de Andahuaylas, siendo las 10:00 horas del día 29 de diciembre del año 2023, se reunieron los docentes: Dr. Enrique Edgardo Condor Tinoco, Dr. Humberto Silvera Reynaga, M.Sc. Jesús Farfán Inca Roca, en condición de integrantes del Jurado Evaluador del Informe Final de Tesis intitulado: "SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL", cuya autora es la Bachiller en Ingeniería de Sistemas **WRITHNEY XIMENA LAZO MARTÍNEZ** y el asesor Msc. Carlos Yinmel Castro Buleje, con el propósito de proceder a la sustentación y defensa de dicha tesis.

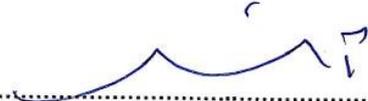
Luego de la sustentación y defensa de la tesis, el Jurado Evaluador **ACORDÓ: APROBAR** por **UNANIMIDAD** a la Bachiller en Ingeniería de Sistemas **WRITHNEY XIMENA LAZO MARTÍNEZ**, obteniendo la siguiente calificación y mención:

Nota escala vigesimal		Mención
Números	Letras	
14	CATORCE	BUENO

En señal de conformidad, se procedió a la firma de la presente acta en 03 ejemplares.


.....
Dr. Enrique Edgardo Condor Tinoco
Presidente del Jurado Evaluador


.....
Dr. Humberto Silvera Reynaga
Primer Miembro del Jurado Evaluador


.....
M.Sc. Jesús Farfán Inca Roca
Segundo Miembro del Jurado Evaluador



APROBACIÓN DEL JURADO DICTAMINADOR

La tesis: **"SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MULTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL"**, para optar el título profesional de ingeniero de sistemas, ha sido evaluado por el Jurado Dictaminador conformado por:

PRESIDENTE: Dr. Enrique Edgardo Condor Tinoco

PRIMER MIEMBRO: Dr. Humberto Silvera Reynaga

SEGUNDO MIEMBRO: Msc. Jesús Farfán Inca Roca

Andahuaylas, 9 de enero del 2024

Dr. Enrique Edgardo Condor Tinoco
Presidente Jurado Evaluador

Dr. Humberto Silvera Reynaga
Primer Miembro Jurado Evaluador

Msc. Jesús Farfán Inca Roca
Segundo Miembro Jurado Evaluador



Universidad Nacional José María Arguedas

Identidad y Excelencia para el Trabajo Productivo y el Desarrollo

Unidad de Investigación de la Facultad Ingeniería
C005-2024

Andahuaylas, 16 de enero de 2024

La Unidad de Investigación de la Facultad de Ingeniería, expide la:

Constancia

De porcentaje de similitud (24%) según el software Turnitin, al informe final de investigación: Sistema para la generación y calificación de cartillas de exámenes de selección múltiple basado en reconocimiento óptico de marcas y visión computacional. Presentado por el Bach. Wrihney Ximena Lazo Martínez cuyo asesor es el MSc. Carlos Yinmel Castro Buleje.

Dra. María del Carmen Delgado Laine
Presidente de la Unidad de Investigación de la
Facultad de Ingeniería

MSc. Fidelia Tapia Tadeo
Miembro de la
Unidad de Investigación de la Facultad de Ingeniería

M.Sc. Richard Carrión Abollaneda
Miembro de la
Unidad de Investigación de la Facultad de Ingeniería

C.c
Archivo.

PAPER NAME

SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN R

AUTHOR

WRITHNEY XIMENA LAZO MARTÍNEZ

WORD COUNT

16053 Words

CHARACTER COUNT

96412 Characters

PAGE COUNT

96 Pages

FILE SIZE

4.5MB

SUBMISSION DATE

Jan 9, 2024 4:09 PM GMT-5

REPORT DATE

Jan 9, 2024 4:10 PM GMT-5

● 10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 9% Internet database
- 2% Publications database
- Crossref database
- Crossref Posted Content database

● Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less than 15 words)



M.Sc. Carlos Yinmel Castro Buleje
ASESOR

● 10% Overall Similarity

Top sources found in the following databases:

- 9% Internet database
- 2% Publications database
- Crossref database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	dspace.unitru.edu.pe Internet	1%
2	repositorio.unajma.edu.pe Internet	<1%
3	coursehero.com Internet	<1%
4	Carlos Yinmel Castro Buleje, Yalmar Temistocles Ponce Atencio, Enriq... Crossref	<1%
5	docplayer.es Internet	<1%
6	tesis.ipn.mx Internet	<1%
7	dspace.otalca.cl Internet	<1%
8	hdl.handle.net Internet	<1%
9	ilamparas.com Internet	<1%

10	digibug.ugr.es Internet	<1%
11	compelect.com.co Internet	<1%
12	secgen.unajma.edu.pe Internet	<1%
13	encorsi.blogspot.com Internet	<1%
14	repositorio.puce.edu.ec Internet	<1%
15	juanbarrios.com Internet	<1%
16	fime.uanl.mx Internet	<1%
17	repositorio.urp.edu.pe Internet	<1%
18	fr.slideshare.net Internet	<1%
19	pdfcookie.com Internet	<1%
20	componentesdelacompu.wordpress.com Internet	<1%
21	soddu2.dst.polimi.it Internet	<1%

22	repositorio.ucv.edu.pe Internet	<1%
23	dspace.unach.edu.ec Internet	<1%
24	mwebs.com.uy Internet	<1%
25	repositorio.espe.edu.ec Internet	<1%

**FORMATO DE AUTORIZACIÓN DE PUBLICACIÓN EN EL REPOSITORIO
INSTITUCIONAL DIGITAL (RIDUNAJMA)**

1. DATOS DEL AUTOR

Nombres y apellidos Writhney Ximena Lazo Martínez
DNI / Carne de Extranjería / 70195801
Pasaporte No.
Correo Institucional writhneym@gmail.com
Teléfono / Celular 931301041

2. GRADO, TÍTULO O ESPECIALIZACIÓN (marque con una X)

Bachiller Licenciado / Título profesional	Magister	Doctor	Segunda especialidad
X			

3. TIPO DE INVESTIGACIÓN (marque con una X)

Tesis	Trabajo Académico	Trabajo de investigación	Trabajo de suficiencia profesional
X			

Título: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MULTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL

Asesor: M.S.c Carlos Yinmel Castro Buleje

Año: 2023

Unidad Académica



Firma

Writhney Ximena Lazo Martínez
DNI: 70195801

LICENCIAS

Licencia estándar

Bajo los siguientes termino, autorizo el deposito de mi trabajo de investigación en el repositorio institucional digital de la UNAJMA.

Con la autorización de deposito de mi trabajo de investigación, otorgo a la Universidad Nacional José María Arguedas, una licencia No exclusiva para reproducir, distribuir, comunicar al público, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del publico mi trabajo de investigación (incluido el resumen), en formato físico o digital, en cualquier medio conocido o por conocerse, a través de los diversos servicios provistos por la universidad, creados o por crearse, tales como el repositorio institucional digital de la UNAJMA, colección de tesis, entre otros, en el Perú y en el extranjero, por el tiempo y veces que considere necesarias, y libre de remuneraciones.

En virtud de dicha licencia, la Universidad Nacional José María Arguedas, podrá reproducir mi trabajo de investigación en cualquier tipo de soporte y en más de un ejemplar, sin modificar su contenido, solo con los propósitos de seguridad, respaldo y preservación.

Declaro que el trabajo de investigación es una creación de mi autoría y exclusiva titularidad, o coautoría con titularidad compartida, y me encuentro facultado a conocer la presente licencia y, asimismo, garantizo que dicho trabajo de investigación es original y no infringe derechos de autor de terceras personas.

La Universidad Nacional José María Arguedas consignara el nombre de/los autor/es del trabajo de investigación, y no les hará ninguna modificación mas de la permitida en la presente licencia.

Autorizo el depósito (marque con una X)

Si autorizo que se deposite inmediatamente

No autorizo

Fecha 29-02-2024



Firma

DNI: 70195801

Writhney Ximena Lazo Martínez

ÍNDICE

CAPITULO I: PROBLEMA DE INVESTIGACIÓN.....	12
1.1. Realidad problemática.....	12
1.2. Formulación del problema.....	13
1.2.1. General.....	13
1.2.2. Específicos.....	13
1.3. Objetivos.....	13
1.3.1. General.....	13
1.3.2. Específicos.....	13
1.4. Justificación.....	14
CAPITULO II: ANTECEDENTES.....	15
2.1. Antecedentes o trabajos previos.....	15
CAPITULO III: MARCO TEÓRICO.....	19
3.1. Definición o marco conceptual.....	19
3.1.1. Pruebas de selección múltiple o pruebas de opción múltiple.....	19
3.1.2. Sistemas de reconocimiento óptico de marcas.....	20
3.1.3. Reconocimiento óptico de marcas (ORM).....	21
3.1.4. Visión computacional.....	21
3.1.5. Procesamiento de imágenes.....	22
3.1.6. Metodología incremental.....	22
3.1.7. Arquitectura monolítica.....	24
3.1.8. Calidad de software.....	25
3.1.9. Precisión y exactitud.....	28
3.1.10. FireBase.....	29
3.1.11. Requerimientos para el desarrollo de software.....	30

3.1.12. Caso de uso	30
3.1.13. Base de datos no relacional.....	31
3.1.14. Figma.....	31
3.1.15. jsQR.....	31
3.1.16. jsPDF.....	32
3.1.17. xlsJS.....	32
3.1.18. QRCode.....	32
3.1.19. OpenCV.js.....	33
3.1.20. Eliminación de ruido.	33
3.1.21. Transformación de perspectiva.	33
3.1.22. Reconocimiento de contornos.	34
3.1.23. Inversión de color.	34
CAPITULO IV: METODOLOGÍA DE LA INVESTIGACIÓN.....	35
4.1. Metodología a usar.	35
4.1.1. Tipo y nivel de investigación.	35
4.1.2. Diseño de investigación.	35
4.1.3. Población.....	35
4.1.4. Tamaño de muestra.	36
4.1.5. Tipo de muestreo.....	36
4.2. Procedimiento de la investigación.....	36
4.2.1. Análisis.....	37
4.2.2. Diseño.....	39
4.2.3. Implementación.....	41
4.2.4. Integración y prueba del sistema.....	53
4.3. Técnicas e instrumentos de recolección de datos.....	54

4.3.1. Objetivo específico 01 – tiempos de procesamiento por cartilla.	55
4.3.2. Objetivo específico 02 - niveles de proporción ancho/alto óptimos para la detección de marcas.	56
4.3.3. Objetivo específico 03 - niveles de umbral óptimo para la correcta detección de marcas.	57
4.3.4. Objetivo general.	58
CAPITULO V: RESULTADOS	61
5.1. Resultado específico 01 – tiempos de procesamiento por cartilla.	61
5.2. Resultado específico 02 - niveles de proporción ancho/alto óptimos para la detección de marcas.	65
5.3. Resultado específico 03 - niveles de umbral óptimo para la correcta detección de marcas.	66
5.4. Resultado general.	67
CAPITULO VI: DISCUSIÓN.....	68
CONCLUSIONES.....	70
RECOMENDACIONES	71
REFERENCIAS BIBLIOGRÁFICAS	72
ANEXOS.....	77

LISTA DE TABLAS

Tabla 1 Requerimientos funcionales y no funcionales	37
Tabla 2 Resumen de tiempo de procesamiento de cartillas.....	64

LISTA DE FIGURAS

Figura 1 Diagrama de caso de uso: usuario/sistema.....	37
Figura 2 Prototipado de inicio de sesión	39
Figura 3 Prototipado del módulo calificación de cartillas de respuesta	40
Figura 4 Diseño de base de datos	40
Figura 5 Arquitectura del sistema.....	41
Figura 6 Desarrollo del módulo I: registro de estudiantes.....	42
Figura 7 Base de datos para el módulo I: registro de estudiantes.....	42
Figura 8 Interfaz del módulo I: registro de estudiantes	43
Figura 9 Desarrollo del módulo II: generación de cartillas de respuesta	44
Figura 10 Interfaz del módulo II: generación de cartillas de respuesta.....	44
Figura 11 Ejemplo 1 de cartilla generada por el módulo II: generación de cartillas de respuesta	44
Figura 12 Ejemplo 2 de cartilla generada por el módulo II: generación de cartillas de respuesta	45
Figura 13 Interfaz del módulo III: calificación de cartillas de respuesta - cargar respuestas	45
Figura 14 Modulo III: calificación de cartillas de respuestas - ejemplo de cartilla marcada	46
Figura 15 Módulo III: calificación de cartillas de respuesta - cartilla con filtro gaussiano	46
Figura 16 Módulo III: calificación de cartillas de respuesta - aplicación de cv.canny()	47
Figura 17 Módulo III: calificación de cartillas de respuesta - transformación de perspectiva.....	47
Figura 18 Módulo III: calificación de cartillas de respuesta - binarización	48

Figura 19 Módulo III: calificación de cartillas de respuesta - fragmento de algoritmo que encuentra burbujas	48
Figura 20 Módulo III: calificación de cartillas de respuesta - reconocimiento de respuestas correctas.....	49
Figura 21 Interfaz del módulo III: calificación de cartillas de respuesta - cargar cartillas	49
Figura 22 Módulo III: calificación de cartillas de respuesta - QR encontrado.....	50
Figura 23 Módulo III: calificación de cartillas de respuesta - fragmento de algoritmo de clasificación de marcas	51
Figura 24 Módulo III: calificación de cartillas de respuesta - cartilla calificada	51
Figura 25 Módulo III: calificación de cartillas de respuesta - interfaz de cartillas procesadas.....	52
Figura 26 Módulo III: calificación de cartillas de respuesta - interfaz de la sección error	53
Figura 27 Módulo III: calificación de cartillas de respuesta - interfaz de la sección configuración.....	53
Figura 28 Despliegue de sistema - inicio de sesión.....	54
Figura 29 Despliegue del sistema - registro de estudiantes.....	54
Figura 30 Fragmento de algoritmo para tomar tiempo de calificación por cartilla.....	55
Figura 31 Resultados de tiempo calificación por cartilla por la consola.....	55
Figura 32 Fragmento de algoritmo para la recopilación de ancho/alto de marcas	56
Figura 33 Fragmento de algoritmo para exportar ancho/alto de marcas	56
Figura 34 Resultados de ancho/alto de marcas por consola	57
Figura 35 Fragmento de algoritmo para recopilación de umbrales	57
Figura 36 Fragmento de algoritmo para exportar umbrales	58

Figura 37 Resultados de umbrales por consola	58
Figura 38 Modelo de matriz de confusión para medir nivel de exactitud y precisión ...	59
Figura 39 Tiempo de calificación individual, 25 preguntas por cartilla.....	62
Figura 40 Tiempo promedio de calificación, 25 preguntas por cartilla.....	62
Figura 41 Tiempo de calificación individual, 50 preguntas por cartilla.....	63
Figura 42 Tiempo promedio de calificación, 50 preguntas por cartilla.....	63
Figura 43 Tiempo de calificación individual, 75 preguntas por cartilla.....	64
Figura 44 Tiempo promedio de calificación, 75 preguntas por cartilla.....	64
Figura 45 Valores de proporciones de marcas.....	65
Figura 46 Valores de umbral en alternativas en blanco.....	66
Figura 47 Valores de umbral en alternativas marcadas	66

RESUMEN

El presente trabajo de investigación aborda la problemática de la generación y calificación de cartillas de respuestas de selección múltiple o cartillas de respuestas de opción múltiple que son parte del proceso de evaluación en diversas instituciones educativas y académicas, se enfoca principalmente en los desafíos computacionales de eficiencia, precisión y exactitud que estos generan. La propuesta desarrollada es un sistema basado en reconocimiento óptico de marcas “Optical Mark Recognition” (OMR) y visión computacional que surge como una propuesta para mejorar el proceso de generación y calificación de cartillas de respuesta de opción múltiple, destacando una solución alternativa, accesible y económica en comparación con los sistemas actuales de procesamiento de pruebas a través de lectores ópticos computacionales.

El estudio se enfoca principalmente en la identificación de la proporción ancho/alto óptima para la detección de marcas en cartillas generadas en impresión convencional, la determinación de umbrales para la detección de burbujas sombreadas considerando respuestas en blanco y respuestas correctas e incorrectas, así mismo, se evalúa los tiempos de procesamiento por cartilla incorporando la lectura de código Quick Response (QR) como método de identificación de las cartillas, estos elementos son clave para evaluar la exactitud, precisión y eficiencia, señalando cómo la solución beneficiaría en automatizar el proceso de generación y calificación de cartillas de respuestas para las instituciones académicas. Finalmente, la investigación propone una solución tecnológica con el potencial de mejorar significativamente los procesos de evaluación en entornos educativos específicos, abordando estos desafíos prácticos y contribuyendo de manera significativa a la calificación de exámenes de forma accesible y económica para las instituciones académicas.

ABSTRACT

The present research work addresses the issue of generating and grading answer sheets for multiple-choice questions, which are part of the assessment process in various educational and academic institutions. It focuses primarily on the computational challenges of efficiency, precision, and accuracy that these processes entail. The proposed solution is a system based on Optical Mark Recognition (OMR) and computer vision, aiming to enhance the generation and grading of multiple-choice answer sheets. It highlights an alternative, accessible, and cost-effective solution compared to current systems that rely on optical readers for test processing.

The study primarily focuses on identifying the optimal width-to-height ratio for detecting marks on conventionally printed answer sheets. It also involves determining thresholds for detecting shaded bubbles, considering both blank and correct or incorrect responses. Additionally, the research evaluates processing times per answer sheet by incorporating Quick Response (QR) code reading as a method for sheet identification. These elements are crucial for assessing accuracy, precision, and efficiency, illustrating how the proposed solution could streamline the process of generating and grading answer sheets for academic institutions. In conclusion, the research suggests a technological solution with the potential to significantly improve assessment processes in specific educational settings, addressing practical challenges and contributing significantly to affordable and accessible exam grading for academic institutions.

CHUMASQA

Kay kunan yachay maskay llamkayqa, achka akllanayuy kutichiy tarjetakunata utaq achka akllanayuy kutichiy tarjetakunata paqarichiypi hinaspa graduacionpi sasachakuytam qawarin, chaykunam imaymana yachay wasikunapi, yachay wasikunapipas chaninchanapaq ruwaypi kachkan, aswantaqa eficiencia nisqapi computacional sasachakuykunapim qawarikun., precisión hinaspa exactitud nisqa chay kaykunaqa paqarichimunku. Kay ruwasqa propuesta huk sistema Óptico Marca Reconocimiento (OMR) chaymanta computadora rikuypi sayasqa kachkan chaymanta huk propuesta hina paqarin allinchaypaq ruwayta achka akllana kutichiy tarjetakuna ruway chaymanta graduación, huk alternativa, yaykuy atiy chaymanta qullqipa kulan pruebawan tupachisqa sistemas de procesamiento nisqakunata, lectores ópticos computacionales nisqawan.

Kay yachayqa astawanqa riqsichikunmi chay relación óptima ancho/altura nisqapi, chaywanmi tarinapaq marcakuna tarinapaq qillqa maytukunapi generasqa imprenta convencional nisqapi, chay determinación de umbrales nisqakuna tarinapaq muyukuna sombreado nisqakuna tarinapaqmi qawarispas ch'usaq kutichiykunata hinaspa allin hinaspa mana allin kutichiykunata, chaynallataq Procesamiento nisqa pachakunatapas sapa tarjeta chaninchasqa kanku algoritmo Quick Response (QR) leeyta churaspa huk método hina tarjetakuna riqsichiypaq Kay elementokuna clave kanku chaninchaypaq exactitud, precisión chaymanta eficiencia, señalaspas imayna solucion beneficiakunman automatizacionpi proceso generacionpi chaymanta calificacionpi tarjetakuna.de yachay wasikunapaq kutichiykuna. Tukuyninpiqa, investigacionqa huk solucion tecnológica nisqatam propone, chaywanmi allintapuni allinchakunman procesos de evaluación nisqakunata, entornos educativos específicos nisqapi, chaywanmi kay sasachakuykunata allichaspa, chaynallataqmi anchata yanapan examenpa calificacionninpi, chaynapi instituciones académicas nisqapaq, chaymi mana qullqillapaqchu.

CAPITULO I: PROBLEMA DE INVESTIGACIÓN

1.1. Realidad problemática.

La evaluación es uno de los elementos clave del proceso de enseñanza y aprendizaje, por el volumen de información que facilita al profesor y por las consecuencias que tiene para el docente, el alumnado, el sistema educativo en que está integrado y la sociedad (López, 2002), para poder evaluar al estudiante se realizan pruebas de las cuales la más tradicional es la elaboración de pruebas escritas, pero estas pruebas presentan problemas entre los cuales esta los tiempos requeridos para preparar las pruebas de evaluación, corregirlas, comunicar los resultados a los estudiantes, y revisar la corrección de las mismas (Kotsiantis, et al., 2010) en ese sentido, en la actualidad muchas instituciones educativas de diversos niveles así como academias pre universitarias realizan actividades como concursos escolares, exámenes de admisión o evaluaciones integrales, por ende, generan un gran volumen de pruebas, para agilizar el proceso de calificación, se usan pruebas de selección múltiple o también denominado opción múltiple con 3 a 5 alternativas, identificadas con las primeras letras del alfabeto (A, B, C, D, E), siendo una de ellas la respuesta correcta. Sin embargo, la corrección manual de las cartillas de respuesta implica la contrata de personal, aumentando la posibilidad de errores humanos y por consiguiente no siempre se puede garantizar los resultados sin fallos o errores. La solución hoy en día utilizada para automatizar la corrección es la adquisición de un dispositivo de lectura óptica de marcas el cual hace uso de cartillas de respuesta especiales en formato específico y de 120 gramos para una rápida, correcta y estandarizada corrección. A pesar de sus beneficios, esta tecnología implica una inversión económica moderada para dotar de equipos y personal calificado para garantizar dicho proceso, la provincia de Andahuaylas no es ajena a estos problemas ya que en la región en los últimos años se ha venido incrementando la necesidad de calificar exámenes de forma automatizada en los diversos centros Pre universitarios así como en eventos académicos y concursos escolares; los cuales

demandan pruebas de corrección automática debido a la gran cantidad de estudiantes que participa, el empleo de este tipo de pruebas genera elevados costos para la instrucción debido a que se necesitan cartillas especiales y un lector de marcas ópticas, frente a este problema, se desarrolló un sistema basado en Reconocimiento Óptico de Marcas (OMR) y visión computacional para poder minimizar costos, generar cartillas en impresión convencional y mejorar el tiempo, exactitud y precisión en la calificación de cartillas de respuestas, una alternativa económica y accesible frente al método convencional usado hoy en día.

1.2. Formulación del problema.

1.2.1. General.

- ✓ ¿Cuáles son los niveles de exactitud y precisión en el proceso de calificación de exámenes con un sistema basado en reconocimiento óptico de marcas y visión computacional?

1.2.2. Específicos.

- ✓ ¿Cuáles son los tiempos de procesamiento por cartilla?
- ✓ ¿Cuáles son los niveles de proporción con respecto ancho/alto óptimos para la detección de marcas?
- ✓ ¿Cuáles son los niveles de umbral óptimo para la correcta detección de marcas?

1.3. Objetivos.

1.3.1. General.

- ✓ Determinar cuáles son los niveles de exactitud y precisión en el proceso de calificación de exámenes con un sistema basado en reconocimiento óptico de marcas y visión computacional.

1.3.2. Específicos.

- ✓ Determinar cuáles son los tiempos de procesamiento por cartilla.

- ✓ Determinar cuáles son los niveles de proporción con respecto ancho/alto óptimos para la detección de marcas.
- ✓ Determinar cuáles son los niveles de umbral óptimo para la correcta detección de marcas.

1.4.Justificación.

La presente investigación se llevó a cabo con el propósito de abordar la creciente demanda de automatizar el proceso de calificación de exámenes de forma económica. El sistema de reconocimiento óptico de marcas y visión computacional posibilita la generación y calificación automatizada de cartillas para exámenes mediante el uso de tecnología OMR y visión computacional utilizando dispositivos convencionales, además de tener como uno de sus objetivos el contribuir con un avance y mejora significativa en las tecnologías similares utilizadas para el procesamiento y calificación de exámenes; mejorando así la eficiencia, precisión y tiempos del procesamiento. El estudio realizado beneficiará diversos centros educativos, así como el centro preuniversitarios e instituciones que organicen eventos académicos que requieran la generación y calificación automatizada de exámenes de opción múltiple en la provincia, región y el país.

CAPITULO II: ANTECEDENTES

2.1. Antecedentes o trabajos previos.

En el artículo titulado “Sistema con reconocimiento óptico de marcas basado en visión artificial para el procesamiento de pruebas de selección múltiple en concursos escolares” se abordó la implementación de un sistema de calificación y verificación de evaluaciones orientado a concursos escolares desarrollado en la ciudad de Andahuaylas, donde obtuvieron una precisión de 97.21% para la calificación del nivel primario y un 99.24% de precisión para el nivel secundario, además de obtener un promedio en tiempo de calificación de 0.83 segundos para la calificación del nivel primario y 0.65 segundos para la calificación del nivel secundario (Castro et al., 2020).

Ware et al. (2019) presentaron su trabajo de investigación en la India titulado “Cost effective optical mark recognition software for educational institutions” donde plantearon un sistema para calificar cartillas de respuestas del tipo burbuja haciendo uso de la herramienta detección de bordes de la librería OpenCV y lenguaje de programación Python con mucha similitud al trabajo de (Sai et al., 2018) la implementación de dicho algoritmo fue propuesta por (Rosebrock, 2016) quien publicó el post “Bubble sheet multiple choice scanner and test grader using OMR, Python and OpenCV” que explica a detalle la implementación de un algoritmo capaz de detectar marcas del tipo burbuja en una cartilla de 5 preguntas utilizando procesamiento de imágenes con la librería OpenCV y el lenguaje de programación Python. (Rosebrock, 2016) en su implementación, utilizó cartillas de 5 preguntas logrando mostrar el porcentaje de acierto y la detección de todas las marcas, sin embargo, no aborda casos en los que se rellena más de una burbuja, así como para las preguntas en blanco que se deben considerarse en un proceso de calificación usual.

En el artículo titulado “An image processing based optical mark recognition with the help of scanner” desarrollado en la ciudad de Kastamonu, realizaron un algoritmo basado en OMR usando Visual Basic, utilizaron un scanner convencional para digitalizar cartillas de respuestas de 3 tipos: tipo I – 100 preguntas, tipo II – 72 preguntas, tipo III – 40 preguntas, se seleccionaron marcas en forma de rectangular y circular. Lograron un tiempo de procesamiento total de 67.179 segundos para 100 cartillas del tipo I, 86.002 segundos para 100 cartillas del tipo II y 47.624 segundos para 50 cartillas del tipo III. Además, lograron una precisión del 100% en la calificación de cartillas del nivel universitario (Belag et al., 2018).

En el artículo titulado “Automated scoring system for multiple choice test with quick feedback” implementaron un algoritmo basado en OMR y OCR una solución que utiliza rectángulos para representar las marcas y una matriz de 7 segmentos para representar adicionalmente el código del estudiante logrando un tiempo de procesamiento para cada cartilla de 0.4 segundos sin anotaciones y 2.27 segundos con anotaciones, el estudio fue aplicado a estudiantes del nivel universitario en Australia (Alomran & Chai, 2018).

Martínez (2017) presento su investigación “Exámenes con corrección automática para la mejora de los sistemas de evaluación” en este proyecto se presenta un sistema de exámenes escritos basado en test de marcas con justificación de respuestas. La digitalización se basa en un escáner convencional, y en algoritmos de Visión por Computador. Los exámenes corregidos se envían a los alumnos de forma automática y personalizada. El sistema presenta importantes ventajas, tanto para el alumno como para el profesor.

Guevara & Lulichac (2016) realizaron su tesis titulada “Desarrollo de un sistema de reconocimiento óptico de marcas para mejorar el procesamiento de encuestas” en la que desarrollan un sistema de reconocimiento óptico de marcas que mejora el procesamiento de encuestas, usando fundamentos, técnicas y algoritmos de Procesamiento Digital de Imágenes (PDI). La investigación apporto una solución versátil a las tareas de lectura y digitalización de

encuestas, siguiendo las fases del PDI, iniciando con la captura de adquisición de imágenes en la que usaron una fotocopidora digital, luego para el pre – procesamiento implementaron algoritmos de realce de contraste, binarización, rotación y reducción de ruido, esta fase les facilito la segmentación, en la que aplicaron un algoritmo ad-hoc, seguidamente llevaron a cabo la fase de descripción en la cual usaron la técnica de descripción de regiones según su área, finalmente en el reconocimiento aplicaron la técnica de reconocimiento estadístico de patrones. El sistema implementado aborda robustez ante la variabilidad de formatos de encuestas con preguntas cerradas y respuestas singulares y/o múltiples, reduciendo el tiempo de procesamiento y aumentando la confiabilidad de la información procesada. Las encuestas que analizaron fueron brindadas por la Cámara de Comercio de la Libertad, con un formato propuesto por los autores de la investigación. Los resultados que obtuvieron fueron satisfactorios, dado que se obtuvo una eficacia de 92.76% y el tiempo promedio de procesamiento por encuesta fue de 2.852 segundos.

En el artículo titulado “Aplicación del reconocimiento óptico de marcas en el proceso de evaluación continua” se abordó la implementación de una aplicación software para reconocimiento de marcas y caracteres manuscritos basada en procesamiento digital de imágenes y redes neuronales, aplicable a la toma del contenido de una encuesta. El manejo de los datos de forma automática evita el tedioso proceso de digitación manual necesario para el archivado de grandes cantidades de documentos. Donde logran implementar un sistema de carácter cooperativo capaz de extraer automáticamente la información de una encuesta que contiene datos de marcas y/o letras mayúsculas y números; el sistema tiene una precisión para reconocimiento de marcas del 100% y del 90% para caracteres; además la velocidad de procesamiento es buena, ya que para la encuesta de validación se puede realizar el reconocimiento de marcas de 6 000 hojas en una hora (568 hojas con el tiempo de escaneo), mientras que el procesado de marcas y caracteres para 630 hojas es posible en el mismo tiempo

314 hojas teniendo en cuenta el tiempo de escaneo. Por otro lado, el bajo costo del sistema implementado, el fácil manejo de su interfaz y los pocos requerimientos que presenta el sistema, hacen de éste una herramienta atractiva para cualquier usuario; teniendo en cuenta lo anterior, se puede afirmar que este sistema constituye un aporte valioso a la Universidad Industrial de Santander (Querol et al., 2014).

CAPITULO III: MARCO TEÓRICO

3.1. Definición o marco conceptual.

3.1.1. *Pruebas de selección múltiple o pruebas de opción múltiple.*

Watters (2015) en el blog hackededucation nos menciona que las pruebas de selección múltiple, comúnmente conocidas como pruebas de opción múltiple, han sido una forma predominante de evaluación en el sistema educativo estadounidense. Su origen se atribuye en gran medida a Frederick J. Kelly, quien propuso este formato a principios del siglo XX con el objetivo de estandarizar y simplificar las pruebas en masa. La necesidad de eliminar la variación en las evaluaciones individuales de los profesores fue un factor clave que impulsó la adopción de este método. Las pruebas de opción múltiple se caracterizan por presentar cuatro o cinco opciones de respuesta, identificadas comúnmente con letras del alfabeto y para el marcado en estas pruebas se selecciona la respuesta considerando la mejor entre las opciones proporcionadas, las pruebas de opción múltiple han evolucionado como una herramienta central en la evaluación educativa, influenciada por la necesidad de estandarización, eficiencia y objetividad en el proceso de calificación.

También Arocena et al. (2010) en su artículo titulado “Evaluación y validación de pruebas parciales de opción múltiple de un curso universitario de primer año”, no dicen que las Pruebas de Opción Múltiple (POM) representan una forma común de evaluación, especialmente cuando se busca una alta representatividad del conocimiento y la objetividad en la calificación de un gran número de estudiantes. Su aplicación permite una corrección rápida y eficiente, además de abordar no solo el conocimiento fáctico, sino también habilidades analíticas y la comprensión de bases teóricas.

Las pruebas de opción múltiple se pueden administrar de varias maneras, incluidas pruebas en papel, pruebas por computadora o mediante tarjetas de respuestas especializadas.

Para poder automatizar la calificación de esta clase de pruebas se hace uso hardware y software especializado, las cartillas deben cumplir ciertas características para su procesamiento como (ORM Solutions, 2023) nos menciona que el peso del papel debe estar entre los 90 a 148 g/m², de igual manera el tamaño del papel debe estar dentro de 5.7 x 15.2 cm a 21.6 x 35.6 cm. Además, que la adquisición de este tipo de cartillas tiene un costo estimado de S/ 0.50 la unidad.

3.1.2. Sistemas de reconocimiento óptico de marcas.

Watters (2015) realiza una descripción sobre la evolución de las máquinas de prueba y tecnologías asociadas, las cuales ha desempeñado un papel crucial en la evaluación masiva de millones de personas, nos menciona que una de las innovaciones destacadas fue la introducción de la tecnología de "marcar sentido" en 1937 por Reynold Johnson, su patente para un "aparato de puntuación" sentó las bases para la máquina de puntuación de pruebas 805 de IBM, lanzada comercialmente en 1937, que permitía detectar marcas de lápiz en hojas de papel y compararlas con una clave de respuestas. Posteriormente, el profesor Everett F. Lindquist de la Universidad de Iowa propuso una idea tecnológicamente diferente en 1955: el "reconocimiento óptico de marcas". En lugar de detectar electrónicamente las marcas de lápiz, su sistema identificaba ópticamente las marcas. Este método, patentado por Lindquist, sentó las bases para Scantron, una empresa fundada en 1972 cuyo nombre se ha vuelto sinónimo de formularios en papel utilizados en pruebas de opción múltiple.

Ambas patentes indican que el número óptimo de opciones para las preguntas de prueba es cinco. Aunque se sugiere la posibilidad de más opciones, cuatro o cinco opciones se han arraigado en las tecnologías educativas, prácticas y políticas educativas. Actualmente se hace uso de lectores ópticos de marcas, que según especificaciones técnicas (ORM Solutions, 2023) el Scatron Opsan 6 escanea con una velocidad de hasta 2,800 hojas por hora, cuenta con un alimentador manual y automático, también existe la posibilidad de clasificar los formularios escaneados en dos contenedores separados, los formularios mal marcados o incompletos se

rechazan y se detienen en la parte superior de la plataforma de salida para su eliminación por el operador, el formulario debe cumplir con ciertas características como peso de (90 - 148 g/m²), tamaño de (5.7 x 15.2 cm - 21.6 x 35.6 cm). Otro modelo es el escáner Insinght 700 que según especificaciones técnicas (ID Soluciones Integradas, 2023) escanea a una velocidad de hasta 6,600 por hora, cuenta con cabezal con capacidad de leer lápiz y tinta (azul o negra), lee formularios de (64mm x 127mm a 229mm x 356mm).

La adquisición de estos equipos tiene un costo que se varía según las necesidades del usuario ya sea velocidad, volumen datos, modelo u otras características, elevando su precio. por ejemplo, el lector Scatron Opsan 6 que se empleó en esta investigación tiene un costo aproximado de \$4800.

3.1.3. Reconocimiento óptico de marcas (ORM).

El Reconocimiento Óptico de Marcas (OMR), como se dice en inglés, Optical Mark Recognition, es “la captura de datos sin reconocimiento manual” (Wordpress, 2013, pág. 1).

Según Bergeron (1998) la tecnología OMR puede leer marcas que se han hecho en posiciones predefinidas (por ejemplo, "burbujas" para ser rellenas, casillas a marcar) en formularios en papel. La tecnología ORM es simple, confiable, rápida y flexible. Sin entrenamiento especial; es necesario utilizar los formularios en papel, y la lectura automática.

3.1.4. Visión computacional.

Según Sucar & Gómez (2011) la visión computacional trata de alguna forma de emular la capacidad que tenemos las personas de adquirir información mediante la visión, a través de las computadoras, de forma que, mediante la interpretación de las imágenes adquiridas, por ejemplo, con una cámara, se puedan reconocer los diversos objetos en el ambiente y su posición en el espacio.

3.1.5. Procesamiento de imágenes.

Sucar & Gómez (2011) nos dice que el objetivo de procesamiento de imágenes es mejorar la calidad de las imágenes para su posterior utilización o interpretación removiendo defectos, problemas de movimiento o desenfoque, también se puede mejorar ciertas propiedades como: color, contraste, estructura, etc.

Otra definición nos la da Flores (2018); nos dice que parte del procesamiento de una imagen es la modificación por medio de programación, de pequeños elementos que la componen llamados píxeles. Habiendo así, dos hechos: la originalidad de la imagen cuando es tomada por primera vez, y luego el resultado de compresión, optimización, filtrado y otros procesos que forman parte del procesamiento de imagen.

Por ejemplo, una técnica de procesamiento de imágenes es la umbralización. La umbralización es una técnica de segmentación simple y eficiente que permite separar los píxeles de una imagen en escala de grises en dos categorías a partir de un valor umbral de intensidad (Triana et al., 2016). También podemos decir la umbralización convierte de imágenes en escala grises a imágenes binarias, donde a cada píxel se le asigna un valor de 0 (fondo) o 1 (primer plano) y el objetivo del umbral es separar los objetos de interés del fondo identificando un valor de umbral que distinga entre los dos obteniendo una imagen binarizada, la binarización juega un papel crucial en el procesamiento de imágenes al permitir un análisis eficiente y facilitar tareas posteriores como el reconocimiento de objetos y la extracción de características (OpenCV, 2023).

3.1.6. Metodología incremental.

El modelo de desarrollo incremental fue propuesto por Harlan Mills en el año 1980, el enfoque actual del desarrollo incremental se centra en diseñar una implementación inicial, presentarla rápidamente para recibir comentarios del usuario y, si la respuesta es positiva, continuar desarrollando versiones adicionales hasta que el sistema cumpla con todas las

necesidades del cliente. El desarrollo incremental ahora es el estándar más común para el desarrollo de sistemas de aplicación webs de comercios electrónicos, web apps prestadoras de servicios, aplicaciones para dispositivos móviles, entre otros (Costanzo, 2020).

Ortiz (2021) nos menciona que el modelo incremental no es necesariamente rígido, es decir, que puede adaptarse a las características de cualquier tipo de proyecto, además nos describe algunas fases del ciclo de desarrollo de software que se pueden tomar en cuenta:

- **Análisis:** Se delimitan y especifican los requerimientos funcionales y no funcionales del producto de software, así como sus restricciones operativas, en esta etapa se encarga de recopilar, analizar y documentar estos requisitos, generando diversos artefactos como documentos, prototipos y arquitecturas beta que sirven como referencia para la siguiente etapa.
- **Diseño:** Se establece la arquitectura de software, esto implica la identificación y descripción de las abstracciones fundamentales del sistema, como componentes e interfaces de conexión a nivel detallado, teniendo en cuenta las tecnologías adecuadas para su desarrollo.
- **Implementación:** Durante esta etapa, los requerimientos se traducen a algoritmo, siguiendo los lineamientos de diseño establecidos previamente. Además, se verifica que cada componente cumple con su especificación, garantizando la calidad del algoritmo desarrollado.
- **Integración y prueba del sistema:** Las unidades del producto se integran para formar un sistema completo, y se llevan a cabo pruebas exhaustivas para asegurar que se cumplan los requerimientos de software, también se detectan y corrigen posibles inconsistencias.

- **Mantenimiento:** Esta etapa implica posibles modificaciones en el funcionamiento, mejoras, extensiones de funcionalidades, corrección de errores y refinamiento continuo para garantizar su relevancia y eficiencia a lo largo del tiempo.

3.1.7. Arquitectura monolítica.

Los primeros programas informáticos utilizaban una arquitectura monolítica, agrupando todo lo relacionado con el sistema dentro del mismo proyecto. Este tipo de arquitectura se caracteriza por: los programas son fáciles de desarrollar, el despliegue y la ejecución del software son muy sencillos, el costo de desarrollo es bajo en comparación con otras arquitecturas. Los problemas de este tipo de arquitectura, como la escalabilidad o la dificultad para los desarrolladores (necesitan entender todo el código de la aplicación) han hecho que este tipo de desarrollo de software deje de ser utilizado en muchos proyectos (aunque su sencillez y bajo coste hace que siga siendo interesante para ciertos proyectos con bajos requerimientos) (ILIMIT, 2020).

Por otra parte, Blancarte (2021) nos dice que, a pesar del estigma asociado con las aplicaciones monolíticas, siguen siendo relevantes y necesarias en ciertos contextos, como sistemas médicos, programas de escritorio y sistemas de ventas y facturación para pequeñas empresas, además las aplicaciones monolíticas pueden ser de una sola clase o de miles, y su definición radica en su capacidad para ser autosuficientes. Aunque pueden parecer menos atractivas en comparación con enfoques más modernos, todavía ofrecen ventajas como facilidad de desarrollo, escalabilidad y autonomía, este tipo de arquitectura cuenta con los siguiente beneficios: fácil de desarrollar y escalar, pocos puntos de fallo y autonomía, rendimiento más rápido, fácil de probar y también nos menciona algunas desventajas como: anclado a un stack tecnológico, escalado monolítico, dificultad para dividir el trabajo en proyectos grandes, necesidad de compilar nuevas versiones para cambios mínimos.

3.1.8. Calidad de software.

Verity (2022) nos menciona que las empresas que buscan llevar a cabo una transformación digital se centran en mejorar sus procesos mediante automatizaciones y nuevos desarrollos, asegurando la calidad en todo momento. En este contexto, se han establecido estándares internacionales, como la norma ISO/IEC 9126 de 2001, diseñada para garantizar la excelencia en el desarrollo de software, la calidad interna y externa del producto, así como en su uso.

Abdud (2012) nos menciona que hablar de calidad del software implica establecer parámetros para definir los niveles mínimos que un producto debe alcanzar. Sin embargo, muchas características del software son cualitativas y difíciles de cuantificar, se han propuesto varios enfoques para establecer factores cualitativos que afectan la calidad del software, como los factores de calidad de McCall y los propuestos por Hewlett-Packard (FURPS). Además, se han realizado intentos de estandarizar los mecanismos de evaluación de calidad del software, incluyendo normas como ISO 9000, el modelo CMM, el estándar IEEE 730:1984, el plan de garantía de calidad del Consejo Superior de Informática MAP y la norma ISO/IEC 9126. La norma ISO 9126 establece un modelo de calidad para la evaluación de productos de software este estándar define seis características básicas de calidad del software donde cada característica se desglosa en subcaracterísticas que permiten una evaluación más detallada.

A continuación, se proporciona algunas características a evaluar según lo establecido por la norma ISO-9126 (Abdud, 2012; Verity, 2022). La funcionalidad evalúa si el software maneja de manera adecuada el conjunto de funciones para satisfacer las necesidades definidas, atributos incluyen la adecuación, exactitud, interoperabilidad, conformidad y seguridad. La confiabilidad se refiere a la capacidad del software para mantener su nivel de ejecución bajo condiciones normales durante un periodo de tiempo establecido incluye el nivel de madurez, tolerancia a fallas y capacidad de recuperación. La usabilidad evalúa el esfuerzo necesario para

que los usuarios utilicen el sistema, considerando la comprensibilidad, facilidad de aprendizaje y operabilidad. Eficiencia: permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos utilizados, abarcando el comportamiento con respecto al tiempo y a los recursos. La mantenibilidad mide el esfuerzo necesario para realizar modificaciones al software, ya sea para corregir errores o incrementar funcionalidades incluye capacidad de análisis, modificación, estabilidad y facilidad de prueba. La portabilidad se refiere a la habilidad del software para ser transferido de un ambiente a otro, se considera aspectos como adaptabilidad, facilidad de instalación, conformidad y capacidad de reemplazo.

Por otra parte, AppMaster (2023) nos menciona algunas pruebas dirigidas específicamente a aplicaciones web como un proceso en el ciclo de vida del desarrollo de software, que se enfocan en garantizar la calidad, funcionalidad, rendimiento, usabilidad y seguridad de las aplicaciones web para proporcionar una experiencia de usuario positiva. Se detallan pruebas dirigidas a aplicaciones web como lo son: las pruebas funcionales que validan las caracterizadas funcionales de la aplicación asegurando que cumplan con los requisitos especificados, pruebas de usabilidad que evalúan la facilidad de uso de la aplicación y mejorar la experiencia en general, pruebas de rendimiento que determinan la capacidad de la aplicación para manejar una carga de trabajo específica, el rendimiento y la utilización de recursos, pruebas de seguridad para identificar vulnerabilidades y garantizar la protección contra posibles amenazas, pruebas de compatibilidad que verifican la capacidad de la aplicación para funcionar en distintos navegadores y dispositivos, sistemas operativos y tamaños de pantalla, pruebas de regresión se realizan después de modificaciones o actualizaciones para asegurar que los cambios no introduzcan nuevos errores y que la funcionalidad existente no se vea afectada.

También existen otros tipos de pruebas que garantizan la calidad del software como lo son las pruebas unitarias de software, también conocidas como “unit testing”, son herramientas esenciales utilizadas por los desarrolladores para validar fragmentos específicos de código

fuente. Estas pruebas buscan asegurar la calidad, funcionamiento y rendimiento del software, contribuyendo a la detección de errores, el correcto desempeño del sistema y el aumento de la confianza con los clientes, generalmente se aplican en fases iniciales del desarrollo del software, antes de integrar fragmentos más grandes en el sistema, estas pruebas pueden ser automatizadas o manuales. Las automatizadas se ejecutan mediante programación previa, permitiendo la repetición y almacenamiento de información para futuros usos. Las manuales requieren intervención humana y son menos comunes debido a su susceptibilidad a errores y menor velocidad de ejecución (Tamushi, 2022).

Por otra parte, las pruebas de integración son esenciales para garantizar el óptimo funcionamiento y la fiabilidad de sistemas informáticos complejos, estas pruebas evalúan la interacción y operación cohesiva de múltiples módulos de software, que son componentes responsables de tareas específicas, a medida que la creación de aplicaciones interconectadas se vuelve común, las pruebas de integración se vuelven un indicador clave del éxito del software. Las pruebas de integración comienzan con pruebas manuales de componentes, donde los módulos se combinan y se evalúan sus relaciones, se pueden automatizar mediante herramientas y pipelines Integración Continua/Entrega o Implementación Continuas (CI/CD), ayuda a abordar la complejidad y los costos (QALified Building Quality, 2023).

Podemos decir también sobre las pruebas de integración que tienen como objetivo validar las interfaces o puntos de integración entre funciones o componentes individuales en un proyecto de desarrollo. Algunas consideraciones clave incluyen la definición de puntos de control que describan criterios de entrada, como la cobertura de prueba y la disponibilidad de entornos de prueba, los desarrolladores o probadores pueden reutilizar activos de prueba existentes creados para pruebas de unidad, adaptándolos según sea necesario (International Business Machines Corporation [IBM], 2021).

3.1.9. Precisión y exactitud.

Para poder determinar la presión y exactitud de un algoritmo podemos hacer uso de la herramienta de matriz de confusión. Esta herramienta es utilizada en machine learning y la estadística para evaluar el rendimiento de un modelo de clasificación (The Data Scools, 2023).

Se representa por una matriz, donde cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real, esta herramienta nos permite ver qué tipos de aciertos y errores está teniendo nuestro modelo a la hora de pasar por el proceso de aprendizaje con los datos. Los cuatro resultados posibles son:

- Verdadero Positivo (VP): el valor real es positivo y la prueba predijo también que era positivo.
- Verdadero Negativo (VN): el valor real es negativo y la prueba predijo también que el resultado era negativo.
- Falso Negativo (FN): el valor real es positivo, y la prueba predijo que el resultado es negativo. Esto es lo que en estadística se conoce como error tipo II.
- Falso Positivo (FP): el valor real es negativo, y la prueba predijo que el resultado es positivo. Esto es lo que en estadística se conoce como error tipo I.

Después de obtener los resultados posibles, podemos evaluar la exactitud y precisión del algoritmo. La exactitud se refiere a la proximidad del resultado a la verdadera medida, en términos estadísticos, la exactitud está vinculada al sesgo de una estimación y se representa como la proporción de resultados verdaderos (verdaderos positivos y verdaderos negativos) dividida entre el número total de casos examinados (verdaderos positivos, falsos positivos, verdaderos negativos, falsos negativos). Por otro lado, la precisión se relaciona con la dispersión de los valores obtenidos en mediciones repetidas de una magnitud, cuanto menor sea la dispersión, mayor será la precisión y se expresa como la proporción de verdaderos

positivos dividida entre todos los resultados positivos (verdaderos positivos y falsos positivos) (Barrios Arce, 2019).

Otra definición nos da Pradillo (2015) menciona que la exactitud y la precisión, son a menudo confundidas como sinónimos, pero poseen diferencias fundamentales en el ámbito de la medición, la exactitud se define por la proximidad entre el valor medido y el valor "verdadero" de la magnitud, evaluando el error de medida, por otro lado, la precisión se centra en la proximidad entre indicaciones repetidas, midiendo la dispersión de mediciones bajo condiciones específicas. La exactitud puede determinarse con una sola medida, mientras que la precisión requiere evaluación a través de mediciones repetidas, no siendo aplicable a una sola medida, la expresión numérica de la precisión involucra medidas de dispersión como la desviación típica, además, la exactitud está relacionada con el sesgo, un error sistemático, mientras que la precisión no se vincula directamente con él.

$$Exactitud = \frac{VP + VN}{VP + FP + FN + VN} \quad (1) \quad Precisión = \frac{VP}{VP + FP} \quad (2)$$

3.1.10. FireBase.

Es una plataforma en la nube, integrada con Google Cloud Platform (GCP), que usa un conjunto de herramientas para la creación y sincronización de proyectos que serán dotados de alta calidad y seguridad, haciendo posible el crecimiento del número de usuarios y dando resultado también a la obtención de una mayor monetización.

Una de las principales características es su base de datos en tiempo real, que permite a los desarrolladores crear y cargar datos para poder crear aplicaciones donde se sincronizan los datos en tiempo real entre múltiples clientes.

Otra característica es su servicio de alojamiento, permite implementar y servir sus aplicaciones web de forma rápida y sin esfuerzo, proporciona hosting en un entorno seguro y confiable (Tamplin, 2016) .

3.1.11. Requerimientos para el desarrollo de software.

El desarrollo de software implica la distinción entre Requisitos Funcionales (RF) y Requisitos No Funcionales (RNF). Los RF son declaraciones sobre los servicios que el sistema proporcionará y cómo responderá a ciertos insumos. Los RNF, por otro lado, no se centran en las funciones específicas del sistema, sino en propiedades como rendimiento y seguridad. Los RF deben ser completos y coherentes, dividir requisitos ambiguos en declaraciones más específicas facilita la implementación, en cuanto a los RNF, abordan cómo el sistema realiza sus funciones y se derivan de restricciones presupuestarias, políticas organizacionales y regulaciones externas (Medium, 2018).

3.1.12. Caso de uso.

Según Internal Business Machines Corporation [IBM] (2021) un caso de uso es un artefacto que describe una secuencia de acciones que resultan en un valor observable. Se dividen en casos de uso empresarial, que representan acciones en el contexto de procesos empresariales, y casos de uso de sistema, que describen acciones llevadas a cabo por un sistema para un actor externo. Los diagramas de casos de uso pueden contener múltiples casos y relaciones entre actores y casos. Los artefactos de actor proporcionan detalles textuales sobre los actores, utilizables en varios diagramas. Las especificaciones de casos de uso comparten el mismo formato de texto enriquecido y pueden incluir contenido basado en plantillas o instrucciones.

El diagrama de casos de uso consta de tres elementos principales: actor, caso de uso y relación, el actor representa a individuos o grupos que interactúan con el software, incluso otros sistemas o equipos, el caso de uso denota una funcionalidad específica del software y se representa con una elipse, puede rodearse opcionalmente con un rectángulo que simboliza los límites del sistema, la relación se ilustra mediante líneas o flechas y representa la interacción

entre actores y casos de uso. El diagrama sirve para visualizar y comprender las interacciones en un sistema de software (PMOinformatica, 2021).

3.1.13. Base de datos no relacional.

Las bases de datos no relacional o que no trabajan bajo el Lenguaje de Consulta Estructurado (NoSQL) son diseñadas para modelos de datos específicos, ofreciendo esquemas flexibles para aplicaciones modernas. Son reconocidas por su facilidad de desarrollo, funcionalidad y rendimiento a escala. Están especialmente adaptadas para aplicaciones modernas como dispositivos móviles, web y juegos, que requieren bases de datos flexibles, escalables y de alto rendimiento. Sus características incluyen flexibilidad en los esquemas, escalabilidad mediante clústeres distribuidos, alto rendimiento optimizado para modelos de datos específicos y funcionalidades específicas para cada tipo de modelo de datos (Amazon, 2023).

3.1.14. Figma.

Figma es una herramienta de prototipado web y editor de gráficos vectoriales basado en la nube. A diferencia de otras herramientas, Figma permite compartir proyectos en tiempo real con equipos, facilitando la colaboración. Aunque opera en la nube, también ofrece la opción de descarga. Similar a Sketch o InVision Studio, destaca por su usabilidad, facilidad para maquetar piezas digitales, exportación en varios formatos, integración de plugins, y la capacidad de sistematizar y compartir diseños en la nube. Sus características incluyen la posibilidad de trabajar con páginas para organizar elementos y una versión gratuita (Centro de Estudios de Innovación diseño y marketing [CEI], 2023).

3.1.15. jsQR.

Es una biblioteca de JavaScript completamente independiente que proporciona una forma sencilla y eficaz de decodificar código QR de archivos de imágenes o secuencias de vídeo. jsQR admite varias fuentes de entrada, incluidos archivos de imágenes, transmisiones

de video y elementos de lienzo, y proporciona una interfaz personalizable para manejar los resultados de la decodificación. La biblioteca es liviana, rápida y fácil de usar, lo que la convierte en una solución ideal para desarrolladores web que necesitan integrar el reconocimiento de código Quick Response (QR) en sus aplicaciones web (Wolfe, 2021).

3.1.16. jsPDF.

Es una biblioteca JavaScript popular que permite la generación de archivos Portable Document Format (PDF) en el navegador. Proporciona un conjunto completo de funciones para crear, modificar y exportar documentos PDF. Con jsPDF, los desarrolladores pueden generar dinámicamente archivos PDF a partir de Lenguaje de Marcas de Hipertexto (HTML), agregar texto, imágenes, formas y tablas, personalizar la apariencia y el diseño del documento e incluso generar informes o facturas complejas (Hall, 2022).

3.1.17. xlsJS.

Es una biblioteca de JavaScript que se utiliza para analizar y leer archivos de Excel (en formato .xls y .xlsx) en un navegador o en un entorno Node.js. Proporciona funcionalidad para extraer datos de archivos de Excel y trabajar con ellos, permite a los desarrolladores usarlo comúnmente en aplicaciones web que requieren el procesamiento de archivos de Excel, como importar datos de hojas de cálculo, generar informes o realizar análisis de datos y proporciona una forma cómoda de interactuar con archivos de Excel mediante JavaScript, eliminando la necesidad de procesamiento del lado del servidor o el uso de herramientas externas (SheetsJS, 2021).

3.1.18. QRCode.

El desarrollador Shim (2015) nos dice que QRCode es una biblioteca de JavaScript para generar código QR. QRCode es compatible con varios navegadores como Chrome, Firefox, Safari, Opera, Mobile Safari, Android y Windows Mobile y usa etiquetas de tabla en el

Document Object Model (DOM). Una de las principales ventajas de QRCode.js es que no tiene dependencias, lo que facilita su integración en proyectos web.

3.1.19. OpenCV.js.

En la página oficial OpenCV (2023) podemos encontrar que openCV.js es una versión de openCV con la cual se puede trabajar en la plataforma web abierta, proporciona una amplia gama de algoritmos de visión por computadora que se pueden aplicar a datos de imágenes y videos. Estos algoritmos incluyen filtrado de imágenes (desenfoque, nitidez, etc.), transformación de imágenes (cambio de tamaño, rotación, etc.), detección y reconocimiento de objetos, extracción de características, contornos y más. Estos algoritmos permiten a los desarrolladores analizar y manipular datos visuales en tiempo real.

3.1.20. Eliminación de ruido.

En el procesamiento de imágenes para poder reducir el ruido y los detalles que dificulten el procesamiento de imágenes se aplica un tipo de filtro gaussiano. El filtro gaussiano funciona aplicando un promedio ponderado a los píxeles de la imagen, con los pesos determinados por una función gaussiana. Esto da como resultado un efecto de desenfoque que puede ayudar a suavizar el ruido y reducir el impacto de las características de pequeña escala en la imagen. El filtro gaussiano se usa comúnmente en visión por computadora, imágenes médicas y varios otros campos (Pratt, 2006).

3.1.21. Transformación de perspectiva.

La fuente oficial OpenCV (2023) nos dice que la transformación de perspectiva en una imagen es un proceso que permite modificar la perspectiva de una imagen, como si se estuviera viendo desde un ángulo diferente. Se utiliza para corregir la distorsión de la imagen debido a la perspectiva o para realizar efectos de manipulación de imágenes, como el enderezamiento de objetos en una imagen. En OpenCV.js, esta transformación se logra mediante funciones como `cv.warpPerspective()` y `cv.getPerspectiveTransform()`.

3.1.22. Reconocimiento de contornos.

Este proceso es útil para una variedad de aplicaciones, como la detección de bordes, el reconocimiento de formas, la segmentación de objetos y la extracción de características. El reconocimiento de contornos en una imagen en OpenCV.js se realiza utilizando la función `cv.findContours()`, esta función encuentra los contornos de los objetos en una imagen binaria o de escala de grises, los contornos son simplemente una secuencia de Picture Element (píxeles) que forman el borde de un objeto en una imagen. Una vez obtenidos los contornos, se pueden realizar diversas operaciones como dibujarlos, calcular su área, perímetro, entre otras (Kaehler & Bradski, 2016).

3.1.23. Inversión de color.

Este proceso es útil para aplicaciones como mejora de imagen, efectos artísticos y detección de objetos, el propósito de la inversión de color es cambiar la apariencia visual de una imagen, lo que puede ser útil en la mejora de la calidad de la imagen, realce de ciertos detalles y detección de objetos específicos en una imagen. La inversión de color en una imagen se puede lograr restando los valores de Red-Green-Blue (RGB) de cada píxel de 255 (OpenCV, 2023).

CAPITULO IV: METODOLOGÍA DE LA INVESTIGACIÓN

4.1. Metodología a usar.

4.1.1. Tipo y nivel de investigación.

Tipo: Aplicada

Nivel: Investigación cuasiexperimental

4.1.2. Diseño de investigación.

El tipo de investigación del presente proyecto es cuasiexperimental aplicada y consta de dos grupos de acuerdo a los objetivos, quedando el diseño de investigación de la siguiente forma:

$$G1 \rightarrow - \rightarrow O1$$
$$G2 \rightarrow X \rightarrow O2$$

Dónde: G₁: Grupo control

G₂: Grupo experimental

X: Es el sistema de reconocimiento óptico de marcas y visión computacional.

-: Es el lector óptico de marcas.

Ya que nuestra investigación es del tipo cuasiexperimental y aplicada se seleccionaron al azar las cartillas de respuesta para medir la exactitud y precisión en la calificación de exámenes, así como los tiempos de procesamiento por cartilla, umbrales y ancho/alto de marcas.

4.1.3. Población.

La población fueron cartillas de respuestas de pruebas de eventos académicos, así como exámenes del nivel pre universitario, siendo esta una población infinita.

4.1.4. *Tamaño de muestra.*

Como la población es una cantidad grande, realizando el muestreo probabilístico con la ayuda de la fórmula se obtuvo el siguiente resultado:

$$n = \frac{Z_a^2 (p \times q)}{e^2} \quad (3)$$

Ecuación 3 - Fórmula para calcular la muestra con población conocida infinita

Donde:

n: tamaño de muestra buscado; e: margen de error

z: nivel de confianza; q: probabilidad en contra; p: probabilidad de éxito

p = 0.5

e = 0.05

z = 1.96

q = 0.5

La muestra fue de 384 cartillas de respuesta.

4.1.5. *Tipo de muestreo.*

Se usó un muestreo de tipo aleatorio simple.

4.2. **Procedimiento de la investigación.**

El presente proyecto tuvo como objetivo principal medir los niveles de exactitud y precisión en la calificación de exámenes para lo cual se desarrolló el “Sistema para la generación y calificación de cartillas de exámenes de selección múltiple basado en reconocimiento óptico de marcas y visión computacional”. El sistema se desarrolló bajo la metodología incremental y desarrollo de software estándar, el cual tiene como objetivo ir agregando nuevas funcionalidades y el sistema fue evolucionando progresivamente de acuerdo a lo deseado.

El “Sistema para la generación y calificación de cartillas de exámenes de selección múltiple basado en reconocimiento óptico de marcas y visión computacional” consta de dos módulos principales: generación de cartillas de respuesta y calificación de cartillas de respuesta; además cuenta con un módulo adicional de registro de estudiantes que facilita el registro de

estudiantes para la generación de cartillas de respuesta y posterior calificación. La totalidad del sistema se encuentra alojada en la web, lo que facilita su acceso y uso para los usuarios.

Para el desarrollo se procedió con los siguientes pasos:

4.2.1. Análisis.

El proceso de desarrollo del sistema empezó realizando entrevista a los encargados del proceso de calificación de exámenes del centro pre universitario, para lo cual se plasmó lo recolectado representándolo en el siguiente Diagrama de Interacción de Usuario (UID) y en una tabla de requerimientos.

Figura 1

Diagrama de caso de uso: usuario/sistema.

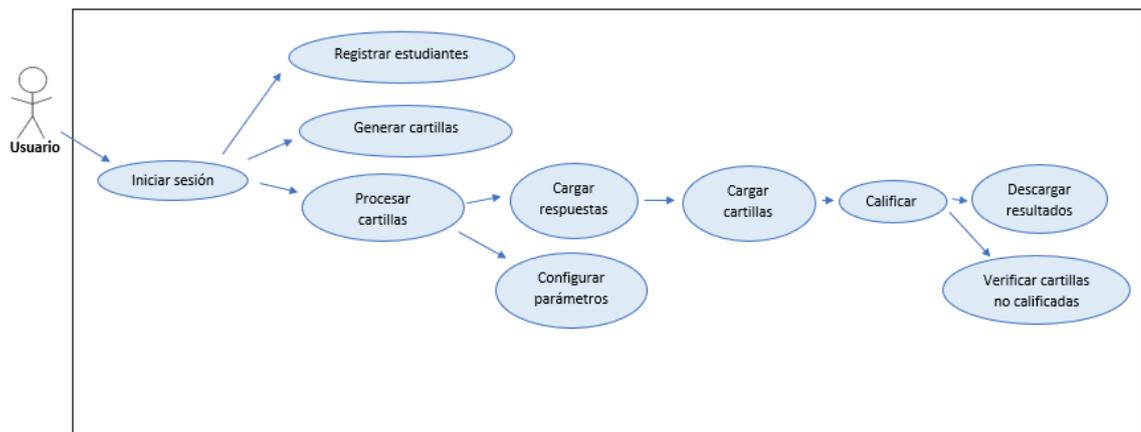


Tabla 1

Requerimientos funcionales y no funcionales.

Requerimientos funcionales	Requerimientos no funcionales
----------------------------	-------------------------------

-
- Registro de estudiantes: el sistema debe permitir el registro de estudiantes (nombres, apellidos, código y grupo).
 - Generación de cartillas: el sistema debe permitir la generación de cartillas con un máximo de 100 preguntas y 5 alternativas.
 - Calificación de cartillas: el sistema debe permitir calificar las cartillas, obtener respuestas (correctas, incorrectas, blancos y puntaje total)
- El sistema debe ser intuitivo y fácil de usar.
 - El sistema debe manejar eficientemente volúmenes de datos.
 - El sistema debe estar disponible un alto porcentaje de tiempo con tiempos de inactividad planificados mínimos.
-

De acuerdo a los requerimientos se desarrolló del sistema empezando con la planificación de los módulos que se integrarían en la plataforma (registro de estudiantes, calificación de cartillas y generación de cartillas), así como las subsecciones y tecnologías que se emplearían. Tras un análisis previo se decidió optar por JavaScript, HTML, Cascading Style Sheets (CSS), OpenCV.js y FireBase, como tecnologías principales para el desarrollo, estas elecciones se basaron en la facilidad de uso de estas tecnologías y en su condición de herramientas de código abierto, además se decidió que la aplicación sea web ya que esto simplifica la experiencia del usuario al eliminar la necesidad de instalaciones y garantiza su disponibilidad prácticamente en su totalidad brindando así una plataforma sólida y accesible para la creación del sistema.

4.2.2. Diseño.

Durante la fase de diseño, se seleccionó la arquitectura del sistema optando por la monolítica, ya que las facilidades que esta ofrece se adecuan perfectamente a las necesidades del sistema, también se llevó a cabo una meticulosa conceptualización de la interfaz y la experiencia de usuario del sistema. Se comenzó por definir los elementos visuales y la disposición de las secciones clave, asegurándose de que la interfaz fuera intuitiva y fácil de navegar. Se elaboraron prototipos que permitieron simular la interacción del usuario con el sistema, identificando posibles mejoras y refinando el diseño visual para lo cual se usó la herramienta Figma.

El diseño y prototipado se realizaron con la finalidad de obtener una representación visual clara de la estructura del sistema, permitiendo validar y ajustar aspectos visuales y funcionales antes de la implementación final. Este enfoque incremental contribuyó significativamente a garantizar que el sistema resultante cumpliera con los estándares de usabilidad y ofreciera una experiencia de usuario intuitiva, además se diseñó una base de datos NoSQL que cumple con las necesidades del sistema en la plataforma firebase.

Figura 2

Prototipado de inicio de sesión.

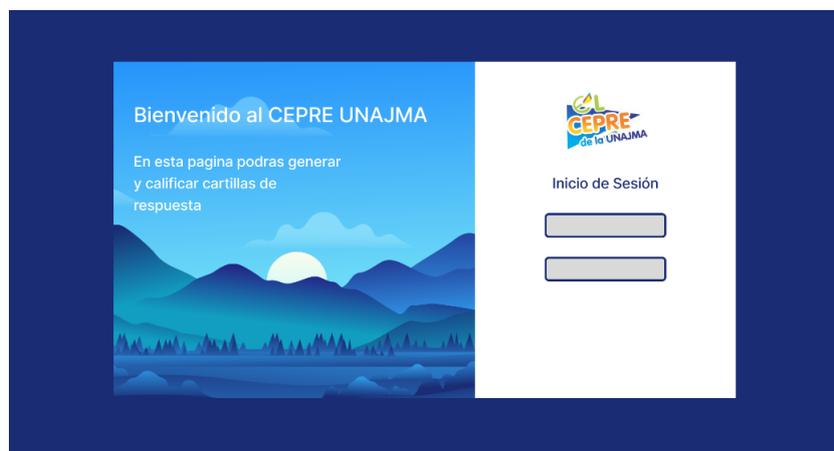


Figura 3

Prototipado del módulo calificación de cartillas de respuesta.



Figura 4

Diseño de base de datos.

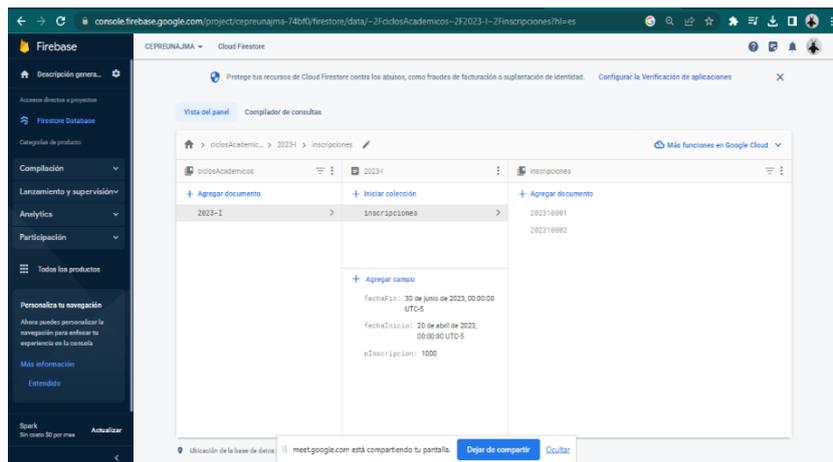
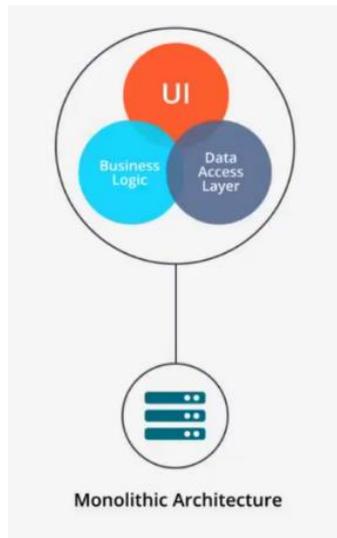


Figura 5

Arquitectura del sistema.



Nota: El grafico representa la arquitectura monolítica, que se eligió para el desarrollo del sistema. Tomado de A. A. Wagner, 2023, de LinkedIn (<https://www.linkedin.com/pulse/arquitectura-monol%C3%ADtica-ariel-alejandro-wagner/?originalSubdomain=es>).

4.2.3. Implementación.

La ejecución de la implementación del diseño y programación se llevó de manera modular siguiendo un enfoque estructurado que asegura una implementación eficiente y coherente del sistema. Los módulos son: registro de estudiantes, generación de cartillas de respuesta y calificación de cartillas de respuesta.

Módulo I: registro de estudiantes.

Se implementó un módulo dedicado al registro de estudiantes, permitiendo la captura y gestión de información personal, académica y relevante para el sistema entre ellas: el código de estudiante, grupo o carrera, apellido paterno, apellido materno y nombres. Este módulo no solo se limitó a la recopilación de datos fundamentales, sino que también ofreció

funcionalidades clave como la capacidad de realizar nuevos registros, editar información existente, buscar y eliminar registros.

Otro aspecto que se puede mencionar es que el módulo de registro de estudiantes se integró con una base de datos en Firebase. Esta conexión permitió un almacenamiento eficiente y seguro de la información recopilada, garantizando la disponibilidad y la persistencia de los datos. La sincronización con Firebase facilitó la gestión dinámica de registros de estudiantes, proporcionando una solución sólida y escalable para el sistema en su conjunto.

Figura 6

Desarrollo del módulo I: registro de estudiantes.

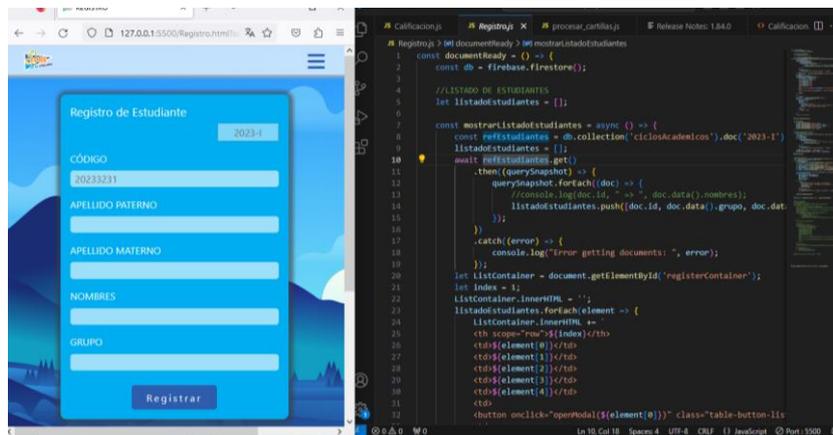


Figura 7

Base de datos para el módulo I: registro de estudiantes.

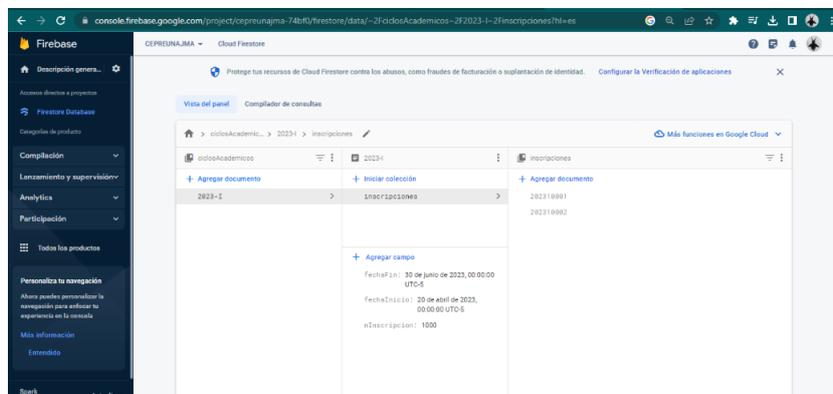


Figura 8

Interfaz del módulo I: registro de estudiantes.



Módulo II: generación de cartillas de respuesta.

Se desarrolló un módulo específico para la generación de cartillas de respuesta en PDF, permitiendo a los usuarios crear y personalizar cartillas según sus necesidades. Se implementaron algoritmos eficientes para generar diseños de cartillas basados en parámetros específicos como: cantidad de preguntas y cantidad de alternativas. Esta implementación facilitó la adaptabilidad del sistema a diversos contextos, permitiendo generar cartillas con un máximo de 100 preguntas y 5 alternativas.

Las cartillas generadas están integradas con la información de los estudiantes, ofreciendo una vinculación directa con los datos del registro, además se ha incorporado un código QR y de barras en cada cartilla, facilitando su identificación precisa para un análisis posterior. Esta funcionalidad no solo mejora la eficiencia en el seguimiento de los datos, sino que también agrega una capa adicional de precisión y accesibilidad al proceso de análisis de las cartillas de respuesta.

Figura 9

Desarrollo del módulo II: generación de cartillas de respuesta.

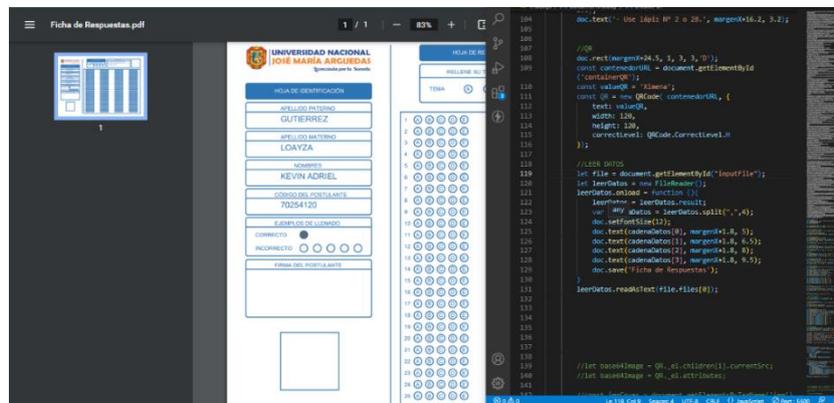


Figura 10

Interfaz del módulo II: generación de cartillas de respuesta.

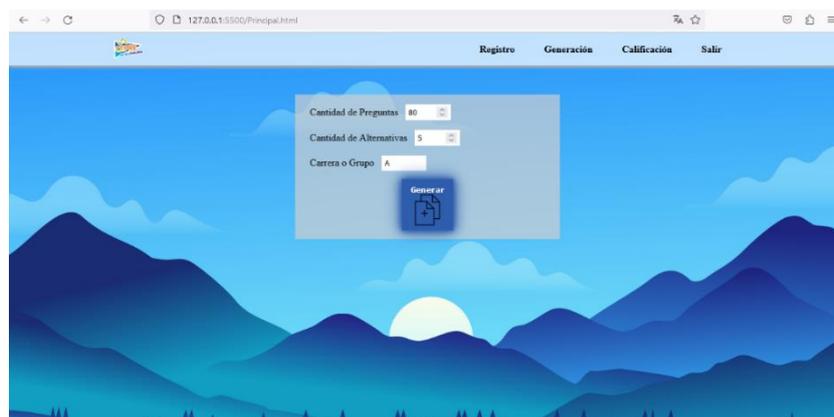


Figura 11

Ejemplo 1 de cartilla generada por el módulo II: generación de cartillas de respuesta.

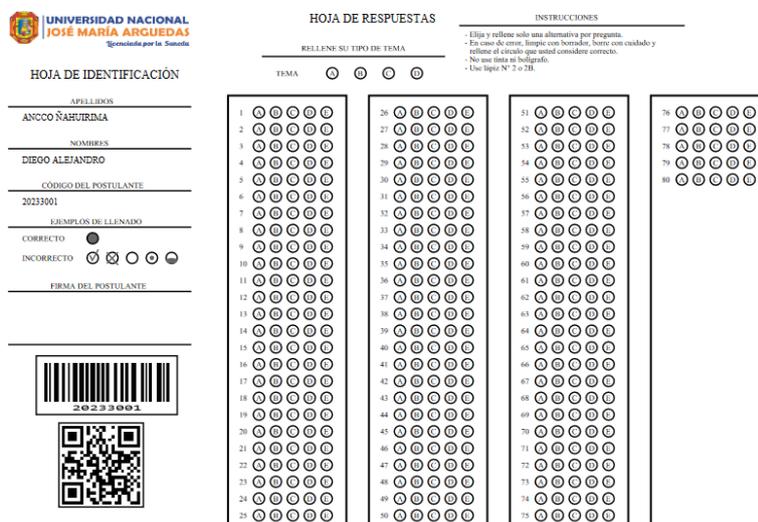


Figura 12

Ejemplo 2 de cartilla generada por el módulo II: generación de cartillas de respuesta.

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS
HOJA DE RESPUESTAS
RELLENE SU TIPO DE TEMA
TEMA ○ ○ ○ ○ ○
INSTRUCCIONES
- Elija y rellene solo una alternativa por pregunta.
- En caso de error, limpie con borrador, borrar con cuidado y rellene el círculo que usted considere correcto.
- No use lápiz ni bolígrafo.
- Use lápiz N° 2 o 2B.

HOJA DE IDENTIFICACIÓN
APELLIDOS
ALARCON VARGAS
NOMBRES
YAQUELIN ANABEL
CODIGO DEL POSTULANTE
20233061
EJEMPLOS DE LLENADO
CORRECTO ●
INCORRECTO ○ ○ ○ ○ ○
FIRMA DEL POSTULANTE

1 ○ ○ ○ ○ ○ 26 ○ ○ ○ ○ ○
2 ○ ○ ○ ○ ○ 27 ○ ○ ○ ○ ○
3 ○ ○ ○ ○ ○ 28 ○ ○ ○ ○ ○
4 ○ ○ ○ ○ ○ 29 ○ ○ ○ ○ ○
5 ○ ○ ○ ○ ○ 30 ○ ○ ○ ○ ○
6 ○ ○ ○ ○ ○ 31 ○ ○ ○ ○ ○
7 ○ ○ ○ ○ ○ 32 ○ ○ ○ ○ ○
8 ○ ○ ○ ○ ○ 33 ○ ○ ○ ○ ○
9 ○ ○ ○ ○ ○ 34 ○ ○ ○ ○ ○
10 ○ ○ ○ ○ ○ 35 ○ ○ ○ ○ ○
11 ○ ○ ○ ○ ○ 36 ○ ○ ○ ○ ○
12 ○ ○ ○ ○ ○ 37 ○ ○ ○ ○ ○
13 ○ ○ ○ ○ ○ 38 ○ ○ ○ ○ ○
14 ○ ○ ○ ○ ○ 39 ○ ○ ○ ○ ○
15 ○ ○ ○ ○ ○ 40 ○ ○ ○ ○ ○
16 ○ ○ ○ ○ ○
17 ○ ○ ○ ○ ○
18 ○ ○ ○ ○ ○
19 ○ ○ ○ ○ ○
20 ○ ○ ○ ○ ○
21 ○ ○ ○ ○ ○
22 ○ ○ ○ ○ ○
23 ○ ○ ○ ○ ○
24 ○ ○ ○ ○ ○
25 ○ ○ ○ ○ ○

Módulo III: calificación de cartillas de respuesta.

Se diseñó un módulo para la calificación automática de cartillas de respuesta mediante el reconocimiento óptico de marcas y visión computacional. La integración de tecnologías claves, como Opencv.js y JavaScript, permitió una calificación precisa y eficiente, reduciendo la carga de trabajo manual. El módulo se estructura en cuatro subpestañas: cargar respuestas, cargar cartillas, errores y configuración.

Primero abordaremos la sección "Cargar Respuestas", en esta sección se permite la carga de la cartilla que contiene todas las respuestas correctas, debidamente marcada y en formato en Joint Photographic Experts Group (JPG) como se ilustra a continuación:

Figura 13

Interfaz del módulo III: calificación de cartillas de respuesta - cargar respuestas.



Figura 14

Modulo III: calificación de cartillas de respuestas - ejemplo de cartilla marcada.

Para iniciar el proceso de reconocimiento de respuestas, el algoritmo empieza con la aplicación de un filtro en escala de grises sobre la cartilla, seguido de la aplicación de un filtro gaussiano que minimiza el posible ruido presente, como pequeñas manchas y líneas que podrían originarse durante el escaneo o llenado de cartillas, como se muestra a continuación.

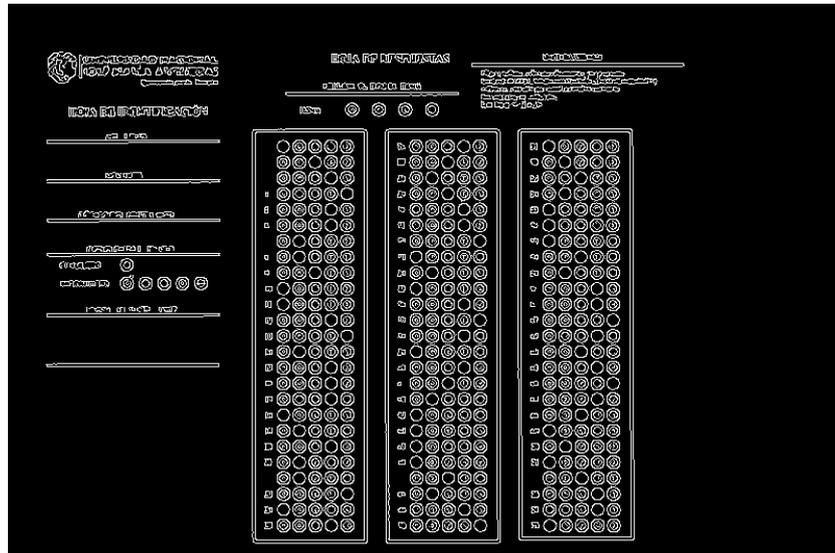
Figura 15

Módulo III: calificación de cartillas de respuesta - cartilla con filtro gaussiano.

Seguidamente se realizó el reconocimiento de contornos por área de cada bloque de preguntas con `cv.findContours` y se guardó en un `cv.MatVector()` para su posterior análisis.

Figura 16

Módulo III: Calificación de cartillas de respuesta - aplicación de `cv.canny`().



Luego de ordenar el `cv.MatVector()` que contiene los bloques de preguntas encontradas se procede a seleccionar el primer elemento, correspondiente al primer bloque de la preguntas del 1 al 25. A continuación, aplicamos una transformación de perspectiva para obtener una vista mejorada del área que será calificada.

Figura 17

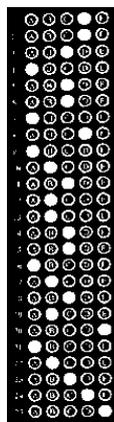
Módulo III: calificación de cartillas de respuesta - transformación de perspectiva.



Antes de proceder a encontrar las burbujas correspondientes a cada alternativa aplicamos la binarización de la imagen a través de `cv.threshold()` y el umbral Otsu que nos permite segmentar el primer plano del fondo de la imagen, posteriormente nos permitirá volver a aplicar técnicas de extracción de contornos para encontrar cada una de las burbujas de la cartilla.

Figura 18

Módulo III: calificación de cartillas de respuesta - binarización.



A continuación volvimos a aplicar el método `cv.findContours()` para encontrar las burbujas que corresponden a cada alternativa, dibujando un cuadrilátero encima de cada burbuja con `cv.boundingRect()` para de esta forma poder manejar los parámetros de ancho/alto que deben cumplir para su mejor clasificación:

Figura 19

Módulo III: calificación de cartillas de respuesta - fragmento de algoritmo que encuentra burbujas.

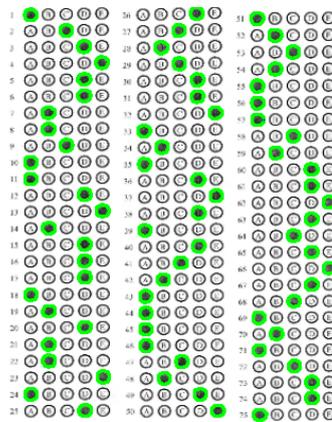
```
//Invertiendo la imagen, aplicando binario invertido y Otsu
let invertido = null;
invertido = new cv.Mat();
cv.threshold(resultado, invertido, 0, 255, cv.THRESH_BINARY_INV | cv.THRESH_OTSU);
let cnts = new cv.MatVector();
//buscando contornos
cv.findContours(invertido.clone(), cnts, hierarchy, cv.RETR_EXTERNAL, cv.CHAIN_APPROX_SIMPLE);

for (let i = 0; i < cnts.size(); i++) {
    let cnt = cnts.get(i);
    let a = cv.boundingRect(cnt);
    let ar = a.width / a.height;
    if (ar >= 0.7 && ar <= 1.7 && a.width >= 9 && a.height >= 9) {
        circulos += 1;
        let p = new cv.Point(a.x, a.y);
        vectorcirculos.push({ punto: p, contorno: cnt });
    }
}
```

Después de ordenar las burbujas encontradas en el bloque de preguntas se procedió a clasificarlas por umbral, si la marca de la burbuja de una fila supera el umbral de las demás de la fila se consideró como la alternativa marcada y aplicando una máscara de color para su identificación, de esta manera se permitió obtener una respuesta para cada pregunta y almacenarlas para su posterior uso, obtuvimos el siguiente resultado:

Figura 20

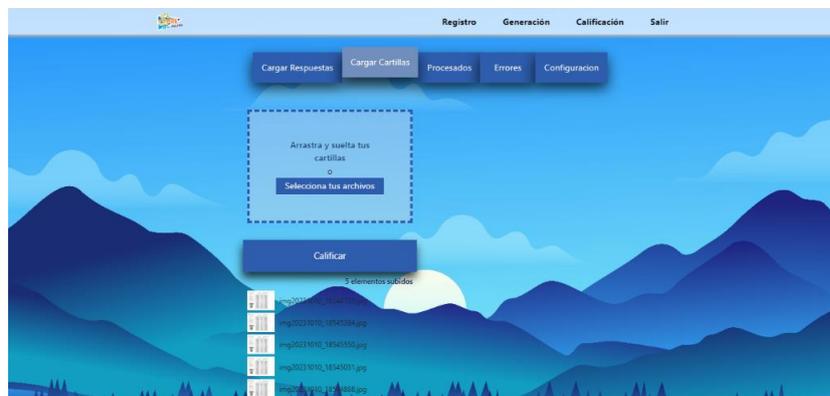
Módulo III: Calificación de cartillas de respuesta - reconocimiento de respuestas correctas.



Al obtener las respuestas correctas de la cartilla se procedió con la calificación de los exámenes, los cuales están en formato JPG y cargados por bloques en la sección "Cargar Cartillas".

Figura 21

Interfaz del módulo III: calificación de cartillas de respuesta - cargar cartillas.



Para iniciar el proceso de calificación de cartillas, el algoritmo también aplica un filtro en escala de grises sobre la cartilla, seguido de la aplicación de un filtro gaussiano que minimiza el posible ruido presente, como pequeñas manchas y líneas que podrían originarse durante el escaneo o llenado de cartilla. Para garantizar la correcta identificación de la cartilla, se comienza reconociendo el código QR que alberga información esencial para el proceso.

La localización de los contornos del cuadrilátero que contiene el QR se realizó mediante la propiedad `cv.findContours()`. Posteriormente, se recortó la parte relevante de la imagen para facilitar su lectura. Utilizando la biblioteca `jsQR`, se procedió a leer la información contenida en el código QR, almacenando la información para el posterior análisis y calificación de la cartilla.

Figura 22

Módulo III: calificación de cartillas de respuesta - QR encontrado.



Seguidamente el algoritmo realizó el mismo procedimiento de reconocimiento de contornos por área de cada bloque de preguntas y después encontrar las burbujas que corresponden a cada respuesta marcada, las cuales fueron almacenadas y comparadas con las respuestas correctas que previamente se analizaron en la sección “Cargar Respuestas”.

Figura 23

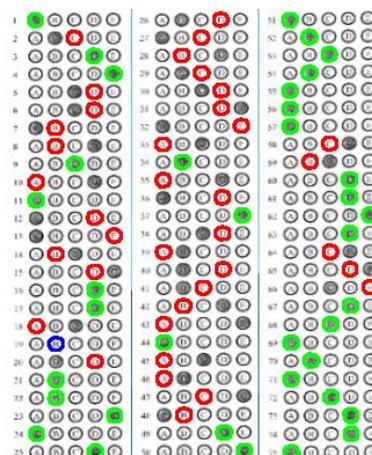
Módulo III: calificación de cartillas de respuesta - fragmento de algoritmo de clasificación de marcas.

```
ANSWER_KEY[e[0]] = bubbled[1]
let k = ANSWER_KEY[e[0]] % 5
let color
if (bubbled[0] > 90) {
  if (k == respuestas[index][1][e[0]][1] && cambios < 2) {
    color = new cv.Scalar(0, 255, 0, 255)
    correct += 1
  }
  else {
    color = new cv.Scalar(255, 0, 0, 255)
    incorrect += 1
  }
}
else {
  color = new cv.Scalar(0, 0, 255, 255)
  blank += 1
}
cv.drawContours([imgColorR, vc, (respuestas[index][1][e[0]][0] * 5) +
  respuestas[index][1][e[0]][1], color, 2]);
})
bloqueRespuestas.push([correct, incorrect, blank]);
resultados.push([imgColorR, bloqueRespuestas])
```

Se llevó a cabo una comparación entre las marcas presentes en la cartilla a calificar y la cartilla con las respuestas correctas. Durante el proceso se sumaron las respuestas correctas, incorrectos y las preguntas en blanco para obtener el puntaje. Se aplicó una máscara de color con el propósito de identificar visualmente las respuestas: aquellas correctas se resaltaron en verde, las incorrectas en rojo, y las preguntas sin respuesta en azul. Obteniendo el siguiente resultado:

Figura 24

Módulo III: calificación de cartillas de respuesta - cartilla calificada.



En la sección "Procesados", se presenta un resumen detallado de las cartillas que han sido calificadas. La información se organiza de manera clara en una tabla que incluye el código del estudiante, las respuestas correctas, las respuestas incorrectas y las respuestas en blanco. Este formato facilita la revisión y el análisis de los resultados de la calificación, proporcionando una visión rápida y estructurada de los datos esenciales, además cuenta con un botón que permite exportar esta información.

Figura 25

Módulo III: calificación de cartillas de respuesta - interfaz de cartillas procesadas.

Código	Correctas	Incorrectas	Blanco
20233132	44	28	3
20233133	22	53	0
20233134	51	23	1
20233135	39	36	0
20233136	12	63	0
20233134	19	56	0
20233118	35	40	0
20233225	39	33	3
20233226	36	36	3
20233211	36	39	0

En caso de que se produzca algún error durante el proceso de reconocimiento de la cartilla, ya sea por dificultades al leer el código QR o al identificar los bloques de preguntas y las burbujas, dichas cartillas no procesadas se muestran en la sección de "Errores". Este apartado sirve para identificar y abordar cualquier inconveniente que haya surgido durante el análisis, permitiendo una identificación efectiva de las cartillas afectadas.

Figura 26

Módulo III: calificación de cartillas de respuesta - interfaz de la sección error.



Concluyendo, la sección de “Configuración” brinda la posibilidad de ajustar diversos parámetros, como el umbral de marcas, la proporción de ancho/alto de las marcas y asignar puntajes específicos a cada respuesta marcada. Estos controles permiten una personalización precisa del sistema, adaptándolo a las necesidades y preferencias específicas del usuario, ofreciendo así una experiencia de calificación más flexible y ajustada a los requisitos particulares.

Figura 27

Módulo III: calificación de cartillas de respuesta - interfaz de la sección configuración.



4.2.4. Integración y prueba del sistema.

En esta fase, se llevó a cabo una verificación del sistema, asegurándose de su correcta operación, entre pruebas de rendimiento, usabilidad, seguridad, compatibilidad,

funcionamiento. Se realizaron la verificación y pruebas de las interfaces, como también pruebas unitarias y de integración como se muestra en (ANEXO 1). Como resultado de este proceso de prueba y corrección, el sistema fue desplegado exitosamente en <https://cepreunajma.firebaseio.com/index.html>.

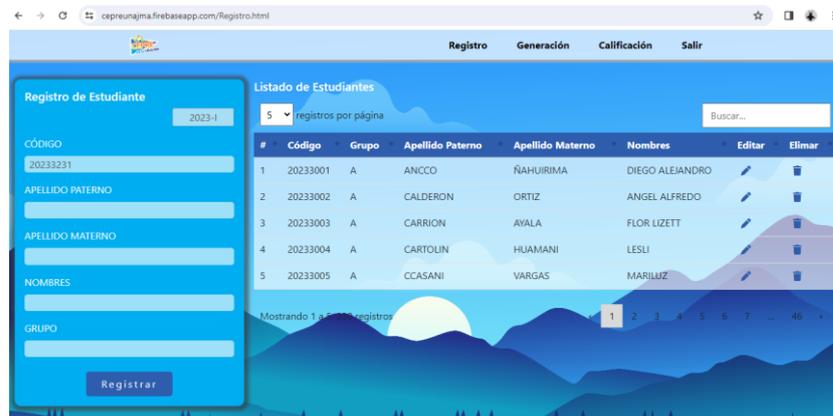
Figura 28

Despliegue de sistema - inicio de sesión.



Figura 29

Despliegue del sistema - registro de estudiantes.



4.3. Técnicas e instrumentos de recolección de datos.

En este apartado se probó el sistema y se detalló como se recopilaban los datos necesarios para alcanzar cada objetivo establecido.

4.3.1. Objetivo específico 01 – tiempos de procesamiento por cartilla.

Con el fin de recopilar el tiempo empleado por el sistema para calificar cada cartilla de respuesta, se registró el tiempo de inicio al cargar los datos de la cartilla dentro de la sección “Cargar cartillas” específicamente en la función “procesarCartillas()” y el tiempo de finalización al obtener los resultados, calculando el tiempo transcurrido mediante la operación: $\text{Tiempo Transcurrido} = \text{Tiempo Fin} - \text{Tiempo Inicio}$. Este procedimiento se implementó según el siguiente fragmento:

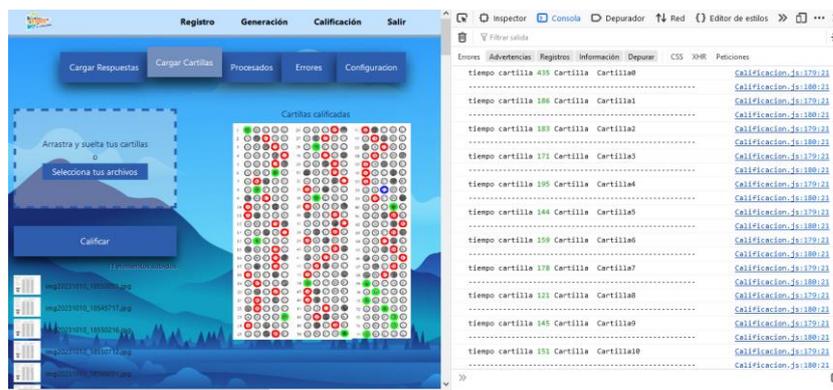
Figura 30

Fragmento de algoritmo para tomar tiempo de calificación por cartilla.

```
//Inicio
let tiempoInicio = new Date().getTime();
procesarCartillas(matcartilla, arrayImágenes);
//Fin
let tiempoFin = new Date().getTime();
let tiempoTranscurrido1 = tiempoFin - tiempoInicio;
console.log('tiempo cartilla', tiempoTranscurrido1, "Cartilla ", element);
console.log("-----")
```

Figura 31

Resultados de tiempo calificación por cartilla por la consola.



Finalmente se registraron los tiempos para cada configuración de cartilla (800x566px, 1600x1132px y 3506x2481px) de forma manual en una tabla detallada que se muestra en el (Anexo 2), (Anexo 3) y (Anexo 4).

4.3.2. Objetivo específico 02 - niveles de proporción ancho/alto óptimos para la detección de marcas.

Con el objetivo de recopilar los niveles de proporción ancho/alto por marcas con en el sistema de calificación para cada cartilla, se almacenaron registros de ancho/alto de marcas en la variable “anchoAlto”, como se muestra en el siguiente fragmento:

Figura 32

Fragmento de algoritmo para la recopilación de ancho/alto de marcas.

```
for (let i = 0; i < cnts.size(); i++) {  
  let cnt = cnts.get(i);  
  let a = cv.boundingRect(cnt);  
  let ar = a.width / a.height;  
  if (ar >= 0.8 && ar <= 1.4 && a.width >= 9 && a.height >= 9) {  
    círculos += 1;  
    let p = new cv.Point(a.x, a.y);  
    vectorcírculos.push({ punto: p, contorno: cnt });  
    //Registro de ancho/alto de marcas  
    anchoAlto.push({ 'ancho': a.width, 'alto': a.height });  
  }  
}
```

Para automatizar la captura de datos, se empleó la librería xlsJS, la cual exportó los datos (ancho/alto de marcas). A partir de estos datos, se calculó la relación de ancho/alto de marcas mediante la fórmula: $\text{Proporción} = \text{Ancho} / \text{Alto}$. Los resultados se muestran en detalle en una tabla adjunta en el (Anexo 5).

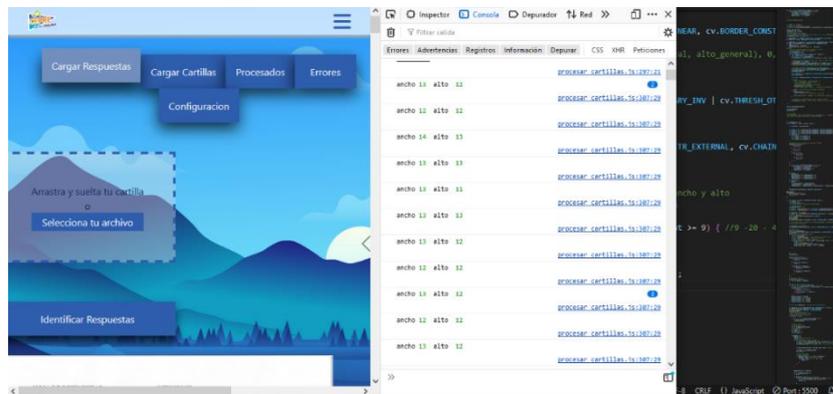
Figura 33

Fragmento de algoritmo para exportar ancho/alto de marcas.

```
const workbook = XLSX.utils.book_new();  
let worksheet = XLSX.utils.json_to_sheet(anchoAlto);  
XLSX.utils.book_append_sheet(workbook, worksheet, 'AnchoAlto');  
XLSX.writeFile(workbook, 'anchoAlto.xlsx');
```

Figura 34

Resultados de ancho/alto de marcas por consola.



4.3.3. Objetivo específico 03 - niveles de umbral óptimo para la correcta detección de marcas.

Con el objetivo de recopilar los niveles de umbral de las burbujas marcadas y burbujas en blanco con en el sistema de calificación para cada cartilla, se almacenaron registros de burbujas marcadas en la variable “umbralesMarcados” y las burbujas en blanco en la variable “umbralesBlancos”, como se muestra en el siguiente fragmento:

Figura 35

Fragmento de algoritmo para recopilación de umbrales.

```
if (total > 90) {
  umbralesMarcados.push({ 'marcadas': total });
}
if (total < 90) {
  umbralesBlancos.push({ 'blanco': total });
}
```

Para automatizar la captura de datos, se empleó la librería xlsJS, la cual exportó los datos (umbrales marcados y umbrales en blanco). Los resultados se muestran en detalle en una tabla adjunta en el (Anexo 6).

Figura 36

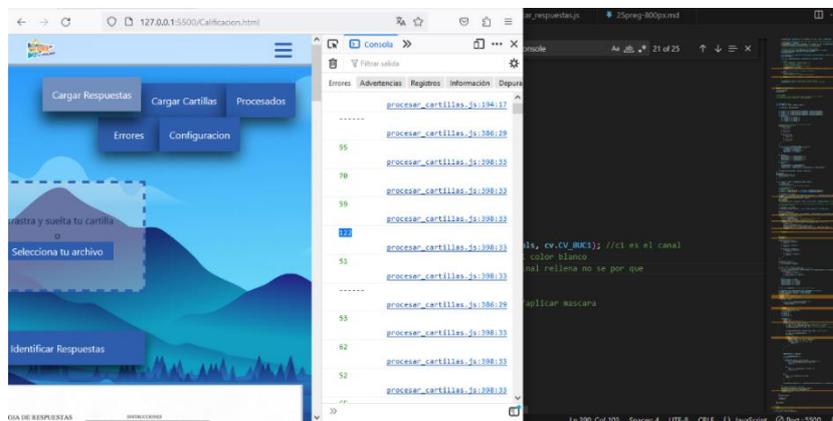
Fragmento de algoritmo para exportar umbrales.

```
const workbook = XLSX.utils.book_new();
let worksheet = XLSX.utils.json_to_sheet(umbralesMarcados);
XLSX.utils.book_append_sheet(workbook, worksheet, 'UmbralesMarcados');
XLSX.writeFile(workbook, 'UmbralesMarcados.xlsx');

const workbook1 = XLSX.utils.book_new();
let worksheet1 = XLSX.utils.json_to_sheet(umbralesBlancos);
XLSX.utils.book_append_sheet(workbook1, worksheet1, 'UmbralesBlancos');
XLSX.writeFile(workbook1, 'UmbralesBlancos.xlsx');
```

Figura 37

Resultados de umbrales por consola.



4.3.4. Objetivo general.

Con el objetivo de evaluar el nivel de exactitud y precisión en el proceso de calificación se realizó la aplicación de la matriz de confusión para lo cual se recopiló los siguientes datos: aciertos positivos en (A, B, C, D, E y pregunta en blanco), fallas en (A, B, C, D, E y pregunta en blanco). Los cuales fueron recolectados de forma manual y tabulados como se muestra a continuación.

Figura 38

Modelo de matriz de Confusión para medir nivel de exactitud y precisión.

	Predicción Alternativa A	Predicción Alternativa B	Predicción Alternativa C	Predicción Alternativa D	Predicción Alternativa E	Predicción Blancos
Valor Real Alternativa A	Aciertos Positivos A	Fallas Falsos B	Fallas Falsos C	Fallas Falsos D	Fallas Falsos E	Fallas Falsos Blancos
Valor Real Alternativa B	Fallas Falsos A	Aciertos Positivos B	Fallas Falsos C	Fallas Falsos D	Fallas Falsos E	Fallas Falsos Blancos
Valor Real Alternativa C	Fallas Falsos A	Fallas Falsos B	Aciertos Positivos C	Fallas Falsos D	Fallas Falsos E	Fallas Falsos Blancos
Valor Real Alternativa D	Fallas Falsos A	Fallas Falsos B	Fallas Falsos C	Aciertos Positivos D	Fallas Falsos E	Fallas Falsos Blancos
Valor Real Alternativa E	Fallas Falsos A	Fallas Falsos B	Fallas Falsos C	Fallas Falsos D	Aciertos Positivos E	Fallas Falsos Blancos
Valor Real Blancos	Fallas Falsos A	Fallas Falsos B	Fallas Falsos C	Fallas Falsos D	Fallas Falsos E	Aciertos Positivos Blancos

Resultados del sistema

	A	B	C	D	E	BL
A	181	0	0	0	0	0
B	0	215	0	0	0	0
C	0	0	291	0	0	0
D	0	0	0	140	0	0
E	0	0	0	0	69	0
BL	0	0	0	0	0	54

$$\text{Precisión en A} = \frac{VP}{VP + FP} = \frac{181}{181 + 5(0)} = 1$$

$$\text{Precisión en B} = \frac{VP}{VP + FP} = \frac{215}{215 + 5(0)} = 1$$

$$\text{Precisión en C} = \frac{VP}{VP + FP} = \frac{291}{291 + 5(0)} = 1$$

$$\text{Precisión en D} = \frac{VP}{VP + FP} = \frac{140}{140 + 5(0)} = 1$$

$$\text{Precisión en E} = \frac{VP}{VP + FP} = \frac{69}{69 + 5(0)} = 1$$

$$\text{Precisión en blancas} = \frac{VP}{VP + FP} = \frac{54}{54 + 5(0)} = 1$$

$$\text{Exactitud} = \frac{VP + VN}{VP + FP + FN + VN} = \frac{181 + 215 + 291 + 140 + 69 + 54}{181 + 215 + 291 + 140 + 69 + 54 + 30(0)} = 1$$

CAPITULO V: RESULTADOS

Se evaluó los niveles de exactitud y precisión en el proceso de calificación de exámenes en distintas circunstancias por medio de los niveles de proporción con respecto ancho/alto de marcas, umbral de marcas y tiempos de procesamiento por cartilla; por medio de un sistema para la generación y calificación de cartillas de exámenes de selección múltiple basado en reconocimiento óptico de marcas y visión computacional en el centro Pre Universitario de la Universidad Nacional José María Arguedas, para tal objetivo se tomó una muestra de 384 cartillas de respuestas, la cual se dividió en dos grupos iguales.

El primer grupo G1 de 192 cartillas fue calificado por un lector óptico de marcas estándar, según especificaciones técnicas (ORM Solutions, 2023) el lector procesa 2000 cartillas en una hora y por ende procesa 1 cartilla en 1800 milisegundos, también se puede ajustar los niveles de sensibilidad de las marcas y separar las cartillas según criterios definidos por el operador y rechazar aquellos mal marcados o incompletos para su posterior revisión.

El segundo grupo G2 de 192 cartillas, fue evaluado en tres aspectos que son tiempo de calificación, la proporción alto/ancho de marcas y umbral de marcas por cartilla, para lo cual se usó el 80% (154) del total de cartillas obteniendo los siguientes resultados:

5.1.Resultado específico 01 – tiempos de procesamiento por cartilla.

Se analizó el tiempo de procesamiento por cartillas mediante el sistema de calificación, dividiendo las cartillas en tres grupos: grupo1 de 25 preguntas, grupo2 de 50 preguntas y grupo3 de 75 preguntas. Cada grupo fue analizado con tres configuraciones distintas de tamaño (800x566px, 1600x1132px y 3506x2481px).

Grupo1 de 25 preguntas: 19 cartillas calificadas.

Figura 39

Tiempo de calificación individual, 25 preguntas por cartilla.

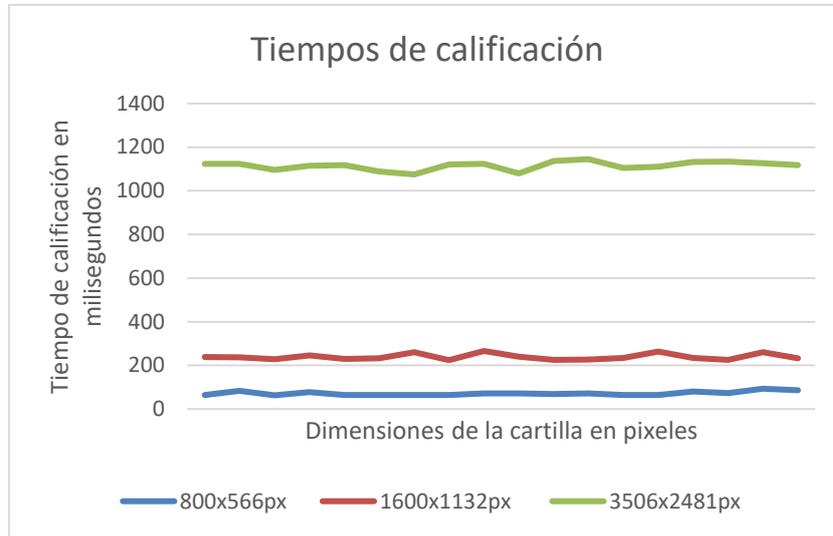
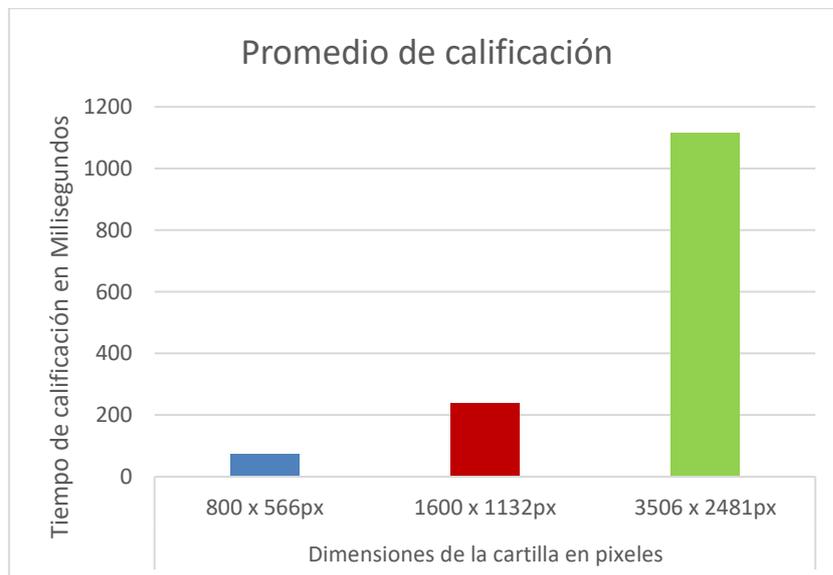


Figura 40

Tiempo promedio de calificación, 25 preguntas por cartilla.



Según la prueba de 19 cartillas con 25 preguntas, se obtuvo un promedio de calificación por cartilla de 71.579 milisegundos para 800x566px, 241.158 milisegundos para 1600x1132px y 1115.263 milisegundos para 3506x2481px.

Grupo2 de 50 preguntas: 60 cartillas calificadas

Figura 41

Tiempo de calificación individual, 50 preguntas por cartilla.

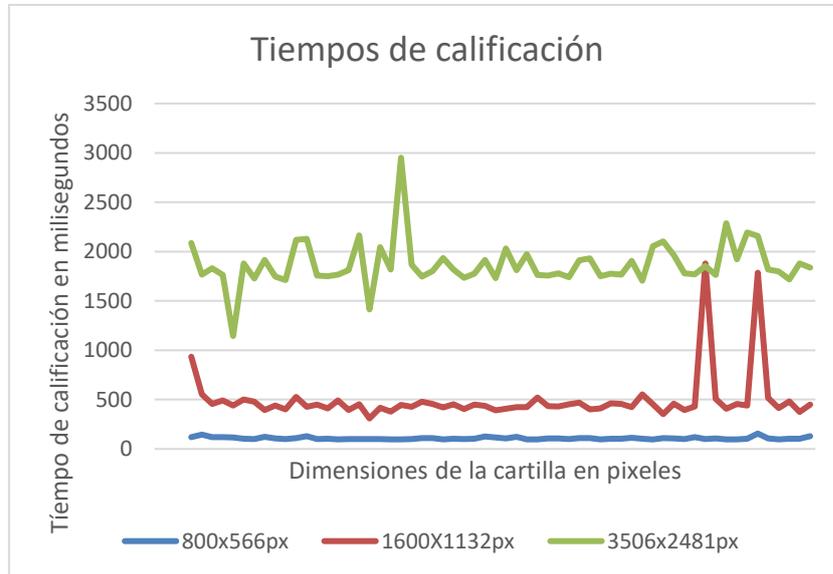
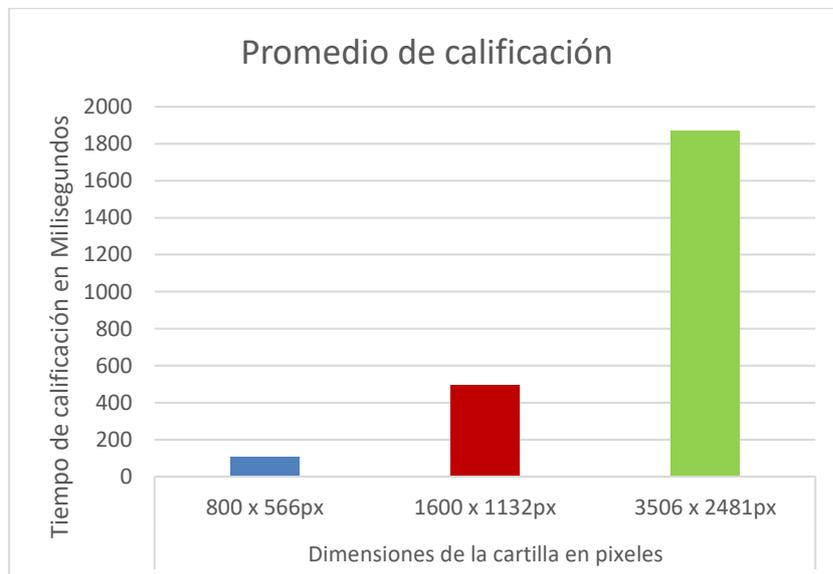


Figura 42

Tiempo promedio de calificación, 50 preguntas por cartilla.



Según la prueba de 60 cartillas con 50 preguntas, se obtuvo un promedio de calificación por cartilla de 107.617 milisegundos para 800x566px, 496.217 milisegundos para 1600x1132px y 1868.517 milisegundos para 3506x2481px.

Grupo3 de 75 preguntas: 75 cartillas calificadas.

Figura 43

Tiempo de calificación individual, 75 preguntas por cartilla.

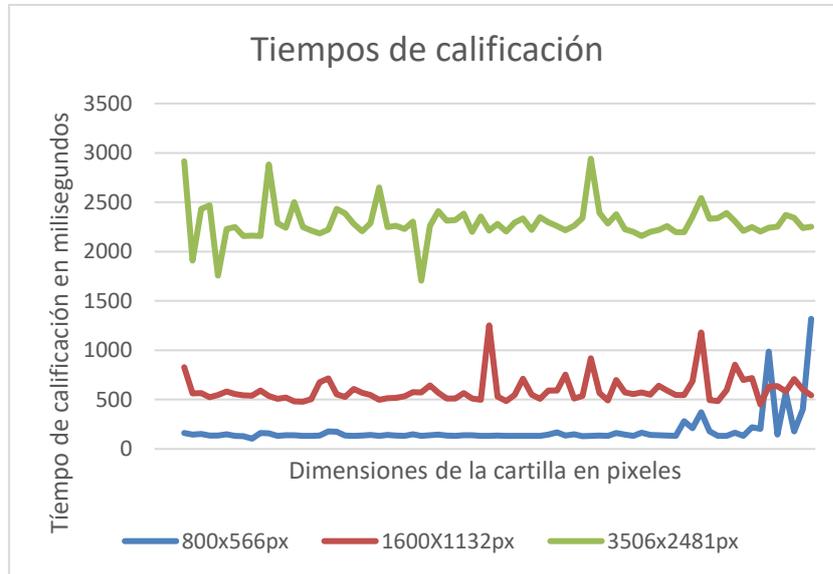
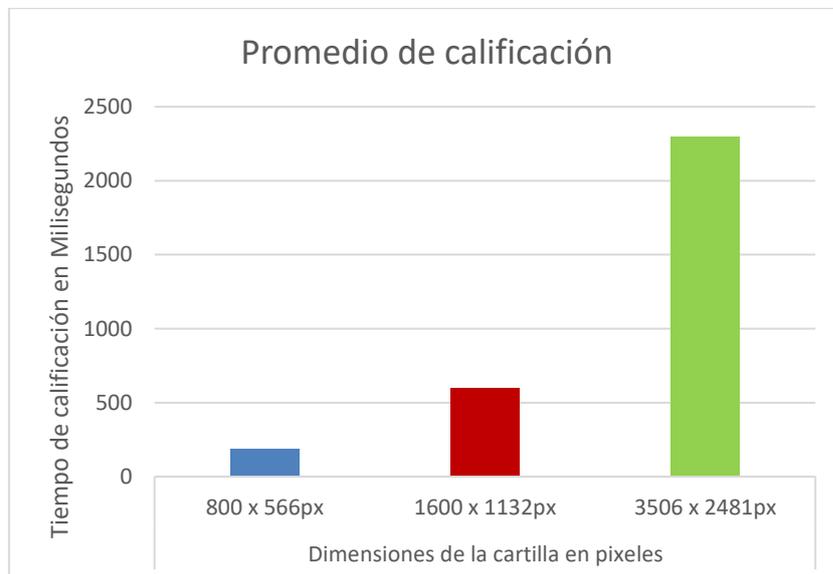


Figura 44

Tiempo promedio de calificación, 75 preguntas por cartilla



Según la prueba de 75 cartillas con 75 preguntas, se obtuvo un promedio de calificación por cartilla de 185.293 milisegundos para 800x566px, 597 milisegundos para 1600x1132px y 2292.946 milisegundos para 3506x2481px.

El resumen de datos obtenidos se muestra en la siguiente tabla.

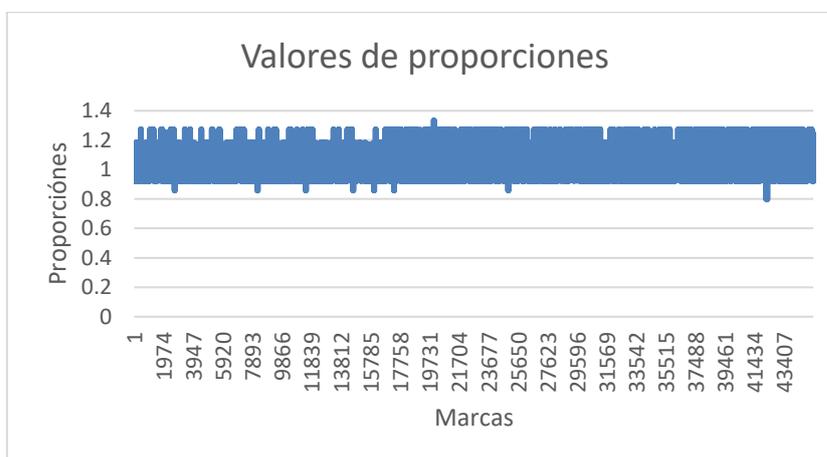
Tabla 2*Resumen de tiempo de procesamiento de cartillas.*

	800x566px	1600x1132px	3506x2481px
Cantidad de preguntas	Promedios en milisegundos		
25	71.579	238.158	1115.947
50	241.158	496.217	1868.517
75	1115.263	597	2292.946

De acuerdo a los resultados obtenidos se determinó que los tiempos de calificación por cartilla más bajos se realizaron con la configuración de 800x566px; por ende, las pruebas de proporción de ancho/alto óptimo y umbral óptimo para la detección de marcas se evaluaron con dicha configuración.

5.2.Resultado específico 02 - niveles de proporción ancho/alto óptimos para la detección de marcas.

Se evaluaron un total de 45375 marcas en 153 cartillas, usando la configuración optima de 800x566px, obteniendo los siguientes datos.

Figura 45*Valores de proporciones de marcas.*

Se determinó que el nivel de proporción ancho/alto de marcas oscila entre 0.8 y 1.333, concluyendo con los valores óptimos como mínimo 0.8 y máximo 1.333.

5.3.Resultado específico 03 - niveles de umbral óptimo para la correcta detección de marcas.

Se evaluaron un total de 36534 burbujas en blanco y 8840 burbujas marcadas en 153 cartillas, usando la configuración optima de 800x566px, obteniendo los siguientes datos.

Figura 46

Valores de umbral en alternativas en blanco.

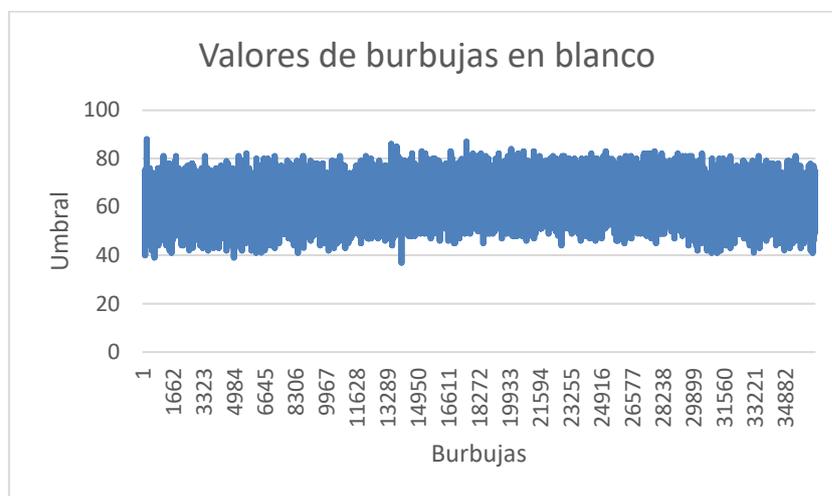
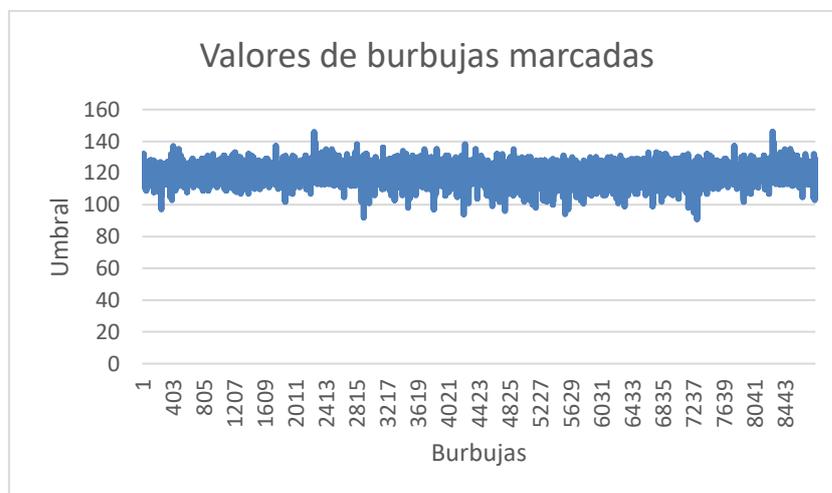


Figura 47

Valores de umbral en alternativas marcadas.



Se determinó que el valor máximo de umbral que puede tomar una burbuja en blanco es 88 y el valor mínimo de umbral que puede tomar una burbuja marcada es de 91.

5.4.Resultado general.

Finalmente, para poder determinar los niveles de exactitud y precisión en la calificación, se usaron las configuraciones optimas de ancho/alto de marcas mínimo 0.8 y máximo 1.333, umbral máximo de burbujas en blanco 88 y umbral mínimo de burbujas marcadas 91 y dimensiones de 800x566px en el 20% (38) de cartillas restantes. Se hizo uso de la Matriz de Confusión, donde la precisión se evalúa para cada posible valor (A, B, C, D, E y pregunta en blanco) y la exactitud usando todos los valores.

Matriz de confusión

	A	B	C	D	E	BL
A	181	0	0	0	0	0
B	0	215	0	0	0	0
C	0	0	291	0	0	0
D	0	0	0	140	0	0
E	0	0	0	0	69	0
BL	0	0	0	0	0	54

Se obtuvo una precisión en A del 100%, precisión en B del 100%, precisión en C del 100%, precisión en D del 100%, precisión en E del 100% y precisión en respuestas en blanco del 100%. En cuanto al nivel de exactitud se obtuvo un nivel de 100%. Se concluye con un nivel de exactitud y precisión del 100% en el proceso de calificación de exámenes con la implementación de un sistema de reconocimiento óptico de marcas y visión computacional.

CAPITULO VI: DISCUSIÓN

Esta investigación tuvo como propósito evaluar los niveles de exactitud y precisión, umbrales óptimos y proporción ancho/alto óptimos de marcas en el proceso de calificación de exámenes en el centro Pre Universitario de la Universidad Nacional José María Arguedas haciendo uso de un sistema para la generación y calificación de cartillas de exámenes de selección múltiple basado en reconocimiento óptico de marcas y visión computacional.

En relación al tiempo de procesamiento por cartilla se realizó una comparación con el sexto antecedente (Guevara & Lulichac, 2016), el tercer antecedente (Belag et al., 2018) y el primer antecedente (Castro et al., 2020).

- En el quinto antecedente, lograron optimizar el tiempo de revisión de encuestas de 39,407 segundos manual a 2,852 segundos promedio por revisión con el sistema.
- En el tercer antecedente se calificaron cartillas de respuesta universitaria de 3 tipos, obtuvieron un tiempo promedio de calificación de 0.672 segundos para el tipo I, 0.860 segundos para el tipo II y 0.952 segundos para el tipo III por cartilla.
- En el primer antecedente, donde se calificaron exámenes de nivel primario y secundario, obtuvieron tiempos promedio de calificación de 0.83 segundos para el nivel primario y 0.65 segundos para el nivel secundario por examen.

Comparando con los resultados que se obtuvieron en el proceso de calificación de exámenes, se logró un promedio de 0.071 segundos por cartilla de 25 preguntas, 0.241 segundos por cartilla de 50 preguntas y 1.115 segundos por cartilla de 75 preguntas. Estos resultados indican una notable reducción del tiempo mediante el uso del sistema basado en reconocimiento óptico de marcas y visión computacional.

En relación al nivel de exactitud y precisión en el proceso de calificación de exámenes, se realizó una comparación con sexto antecedente (Querol et al., 2014), el quinto antecedente (Belag et al., 2018) y el primer antecedente (Castro et al., 2020).

- En el sexto antecedente se obtuvo una precisión para el reconocimiento de marcas del 100% y 90% para el reconocimiento de caracteres.
- En el tercer antecedente se obtuvo una precisión del 100% en la calificación de exámenes universitarios de tipo I, II y III.
- En el primer antecedente se obtuvo una precisión en la calificación de exámenes del 97.21% para el nivel primario y un 99.24% de para el nivel secundario.

Comparando con los resultados que se obtuvieron en el proceso de calificación de exámenes, se logró un nivel de exactitud del 100% y nivel de precisión del 100%. Esta mejora es notable en comparación con el primer antecedente, y muestra similitudes con el nivel de precisión logrado en el sexto y quinto antecedente. Estos resultados destacan el nivel de exactitud y precisión del sistema basado en reconocimiento óptico de marcas y visión computacional.

Adicionalmente se incorporaron campos de análisis, como el nivel ancho/alto de marcas y el nivel de umbral de marcas. Estas características, que no fueron previamente abordadas en investigaciones anteriores, podrían representar un nuevo enfoque de estudio en la evaluación de exámenes mediante la combinación de reconocimiento óptico de marcas y visión computacional.

CONCLUSIONES

Se logro determinar cuáles son los tiempos de calificación de cartillas implementando el sistema basado en reconocimiento óptico de marcas y visión computacional, logrando un promedio de calificación de 0.071 segundos por cartilla de 25 preguntas, 0.241 segundos por cartilla de 50 preguntas y 1.115 segundos por cartilla de 75 preguntas.

Se logro determinar cuáles son los niveles de proporción de ancho/alto óptimos para las marcas concluyendo con una proporción mínima de 0.8 y máxima de 1.3.

Se logro determinar cuáles son los niveles de umbral óptimo para las marcas concluyendo que el valor máximo de umbral que puede tomar una burbuja en blanco es 88 y el valor mínimo de umbral que puede tomar una burbuja marcada es de 91.

Se logro determinar cuáles son nivel de exactitud y precisión en la calificación cartillas mediante la implementación del sistema basado en reconocimiento óptico de marcas y visión computacional obteniendo un nivel de exactitud al 100% y nivel de precisión al 100%.

RECOMENDACIONES

- ✓ Se sugiere optar por la impresión a tinta, ya que esto disminuirá la probabilidad de puntos en blanco que pueden aparecer al utilizar tóner en el código QR, evitando posibles complicaciones en su lectura, además de que la impresión a tinta suele ser más resistente a los borradores que puedan realizar los estudiantes al momento del marcado de respuestas.
- ✓ Se sugiere emplear lápiz 2B para completar las cartillas, ya que esto contribuye a la estandarización del umbral de marcas y asegura una calificación precisa de los exámenes.
- ✓ Se recomienda capacitar al personal encargado del proceso de calificación de exámenes para el adecuado uso del sistema de reconocimiento óptico de marcas y visión computacional.
- ✓ Se aconseja calificar bloques de cartillas con un máximo de 50 elementos. Esto previene posibles inconvenientes, como congelamientos de pantalla u otros problemas relacionados con la falta de espacio en memoria, especialmente en equipos con capacidades de RAM más limitadas.

REFERENCIAS BIBLIOGRÁFICAS

Abud Figueroa, M. A. (2012). Calidad en la industria del software. La norma ISO-9126.

UPIICSA.

Alomran, M., & Chai, D. (2018). Automated Scoring System for Multiple Choice Test with

Quick Feedback. *International Journal of Information and Education Technology.*

Amazon. (2023). Obtenido de <https://aws.amazon.com/es/nosql/>

AppMaster. (2023). Obtenido de <https://appmaster.io/es/blog/pruebas-de-aplicaciones-web#diferentes-tipos-de-pruebas-de-aplicaciones-web>

Arocena, R., Gascue, C., & Leymonié, J. (2010). Evaluación y validación de pruebas parciales de opción múltiple de un curso universitario de primer año. *Revista Electrónica de Enseñanza de las Ciencias.*

Barrios Arce, J. I. (2019). *Health Big Data*. Obtenido de <https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/>

Belag, I. A., Gultepe, Y., & Elmalı, T. M. (2018). An Image Processing Based Optical Mark Recognition With the Help of Scanner. *International Journal of Engineering Innovation & Research.*

Bergeron, B. P. (1998). Optical mark recognition. *Postgraduate Medicine online*, 4.

Blancarte, O. (2021). *Oscar Blancarte software Architect*. Obtenido de <https://reactiveprogramming.io/blog/es/estilos-arquitectonicos/monolitico>

Castro, C. Y., Ponce, Y. T., & Condor, E. E. (2020). Sistema con Reconocimiento Óptico de Marcas Basado en Visión Artificial para el Procesamiento de Pruebas de Selección Múltiple en Concursos Escolares. *2020 XLVI Latin American Computing Conference (CLEI).*

- Centro de estudios de innovación diseño y marketing. (2023). Obtenido de <https://cei.es/que-es-figma/>
- Costanzo, M. (2020). *Mwebs*. Obtenido de <https://mwebs.com.uy/blog/qu%C3%A9-es-el-desarrollo-incremental/23>
- Flores, A. C. (2018). *Evolución del Procesamiento Digital de Imágenes*. México.
- Guevara Honores, S. E., & Lulichac Rojas, E. (2016). *Desarrollo de un sistema de reconocimiento óptico de marcas para mejorar el procesamiento de encuestas*. Trujillo.
- Hall, J. (2022). *raw.githack.com*. Obtenido de <https://raw.githack.com/MrRio/jsPDF/master/docs/index.html>
- ID Soluciones Integradas. (2023). Obtenido de <https://www.idsolucionesintegradas.com/>
- ILIMIT. (2020). Obtenido de <https://www.ilimit.com/blog/arquitecturas-monoliticas-o-arquitectura-de-microservicios-ventajas-e-inconvenientes/>
- International Business Machines Corporation (IBM). (2021). Obtenido de <https://www.ibm.com/docs/es/engineering-lifecycle-management-suite/lifecycle-management/6.0.3?topic=requirements-defining-use-cases>
- International Business Machines Corporation (IBM). (2021). Obtenido de <https://www.ibm.com/docs/es/rtw/9.0.1?topic=phases-integration-testing>
- Kaehler, A., & Bradski, G. (2016). *Learning OpenCV 3*. O'Reilly Media, Inc.
- Kotsiantis, S., Patriarcheas, K., & Xenos, M. (2010). A combinational incremental ensemble of classifiers as a technique for predicting students' performance in distance education.
- López, J. M. (2002). *La evaluación en la Universidad. La evaluación del aprendizaje de los alumnos universitarios*.

Martínez Castro, A. (2017). *Exámenes con corrección automática para la mejora de los sistemas de evaluación*. Granada.

Medium. (2018). Obtenido de <https://medium.com/@requeridosblog/requerimientos-funcionales-y-no-funcionales-ejemplos-y-tips-aa31cb59b22a>

OpenCV. (2023). *OpenCV*. Obtenido de <https://docs.opencv.org/3.4/index.html>

ORM Solutions. (2023). *ORM Solutions*. Obtenido de http://www.omrsolutions.com/omr_readers_scanners/scantron/opscan/6/index.php

Ortiz Negrín, D. A. (2021). El modelo incremental para el desarrollo de un sistema informático de gestión de información.

PMOinformatica. (2021). Obtenido de [http://www.pmoinformatica.com/2021/02/elementos-diagrama-casos-de-uso.html#:~:text=Como%20se%20ha%20mencionado%20anteriormente,la%20relaci%C3%B3n%20\(o%20asociaci%C3%B3n\)](http://www.pmoinformatica.com/2021/02/elementos-diagrama-casos-de-uso.html#:~:text=Como%20se%20ha%20mencionado%20anteriormente,la%20relaci%C3%B3n%20(o%20asociaci%C3%B3n)).

Pradillo, B. (2015). *Orbitales Moleculares*. Obtenido de <https://www.orbitalesmoleculares.com/precision-y-exactitud/>

Pratt, W. K. (2006). *Digital Image Processing: PIKS Scientific Inside*. Wiley-Interscience.

QAlified Building Quality. (2023). *QAlified Building Quality*. Obtenido de <https://qalified.com/es/blog/pruebas-de-integracion-que-son/>

Querol, A. E., Benedito, J. L., & Ljiljana, M. (2014). Aplicación del Reconocimiento Óptico de Marcas en el proceso de evaluación continua. 7.

Raeburn, A. (2023). Obtenido de <https://asana.com/es/resources/accuracy-vs-precision>

Rosebrock, A. (2016). *Pyimagesearch*. Obtenido de <https://www.pyimagesearch.com/2016/10/03/bubble-sheet-multiple-choice-scanner-and-test-grader-using-omr-python-and-opencv/>

Sai, Y. S., Srinivas, A. S., & Krishna, G. M. (2018). OMR Evaluation using Image Processing. *International Journal of Innovations & Advancement in Computer Scienc.*

SheetsJS. (2021). *npm*. Obtenido de <https://www.npmjs.com/package/xlsx>

Shim, D. (2015). *github*. Obtenido de <https://davidshimjs.github.io/qrcodejs/>

Sucar, L. E., & Gómez, G. (2011). *Visión Computacional*. México; Alemania.

Tamplin, J. (2016). *The firebase Blog*. Obtenido de <https://firebase.blog/posts/2016/05/firebase-expands-to-become-unified-app-platform>

Tamushi. (2022). *Testing IT*. Obtenido de <https://www.testingit.com.mx/blog/pruebas-unitarias-de-software#:~:text=Una%20prueba%20unitaria%20de%20software,funci%C3%B3n%20de%20proceso%20o%20actividad%20espec%C3%ADfica>.

The Data Schools. (2023). *The Data Schools*. Obtenido de <https://thdataschools.com/python/matriz-confusion/>

Triana, N., Jaramillo, A. E., Gutiérrez, R. M., & Rodríguez, C. A. (2016). Técnicas de umbralización para el procesamiento digital de imágenes de GEM-Foils. *redalyc*.

Universidad Nacional de Trujillo. (2012). *SEACE*. Obtenido de <http://zonasegura.seace.gob.pe/mon/docs/procesos/2012/001894/997562759radC8B0C.pdf>

Universidad Nacional Mayor de San Marcos. (2013). *SEACE*. Obtenido de <http://zonasegura.seace.gob.pe/mon/docs/procesos/2012/001906/1056129552rad6A923.pdf>

Verity. (2022). *Verity*. Obtenido de <https://www.verity.cl/que-es-norma-iso-iec-9126-2001/#:~:text=El%20est%C3%A1ndar%20ISO%2D9126%20establece,trav%C3%A9s%20de%20un%20conjunto%20de>

Ware, V., Menon, N., Varute, P., & Dhannawat. (2019). Cost effective optical mark recognition software for educational institutions. *International Journal of Advance Research Ideas and Innovations in Technology*.

Watters, A. (2015). *Hackededucation*. Obtenido de <https://hackededucation.com/2015/01/27/multiple-choice-testing-machines>

Wolfe, C. (2021). *github*. Obtenido de <https://github.com/cozmo/jsQR#jsqr>

Wordpress. (2013). *WordPress.com*. Recuperado el 28 de 07 de 2019, de WordPress.com: <https://componentesdelacompu.wordpress.com/2013/03/20/reconocimiento-optico-de-marcas-omr-reconocimiento-optico-de-caracteres-ocr-reconocimiento-de-caracteres-mediante-tinta-magnetica-micr-identificacion-por-radiofrecuencia-rfid-etiquetas-de/>

ANEXOS

ANEXO 01 CALIDAD DE SOFTWARE

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

PRUEBAS DE FUNCIONALIDAD

Figura 48

Pruebas de inicio de sesión.



Figura 49

Pruebas de registro de estudiantes.

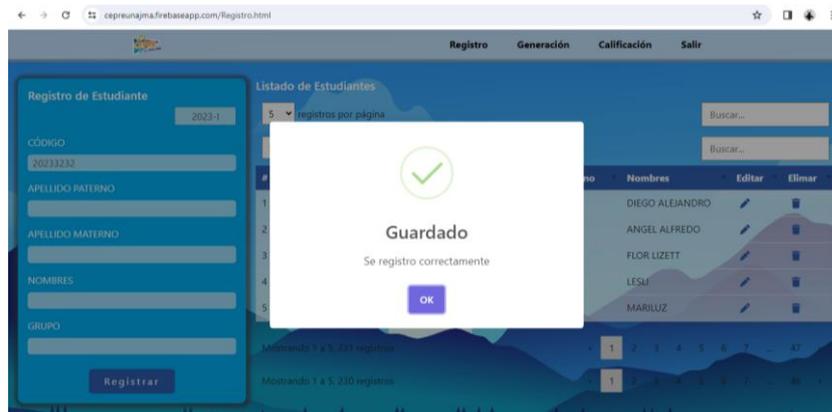


Figura 53

Pruebas de calificación de cartillas, reconocimiento de respuestas correctas.

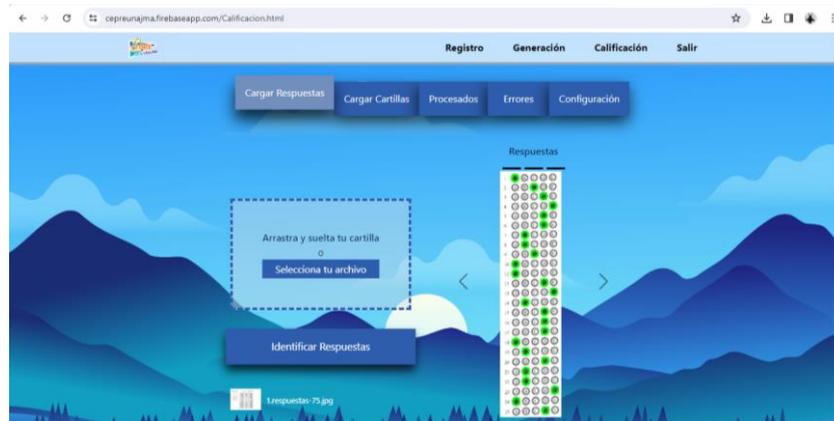


Figura 54

Pruebas de calificación de cartillas, ingreso de cartillas a calificar.

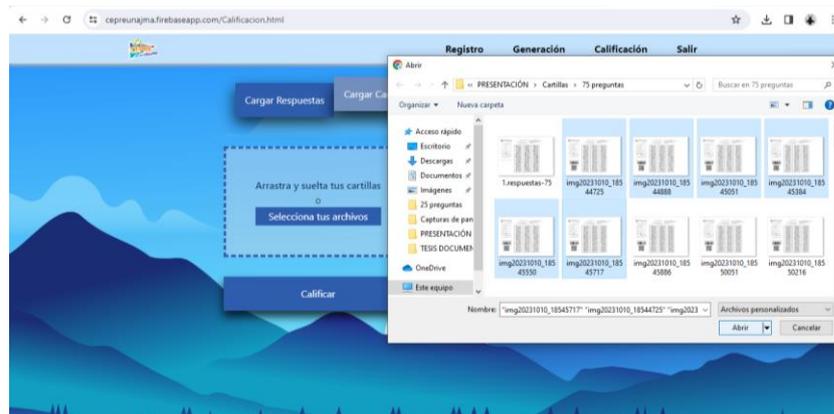
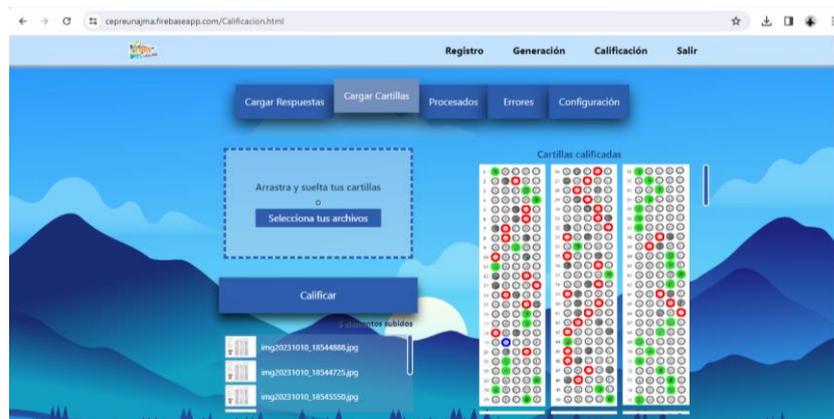


Figura 55

Pruebas de calificación de cartillas, calificación de cartillas.



PRUEBAS DE RENDIMIENTO

Figura 56

Pruebas de rendimiento en la calificación de 10 cartillas.

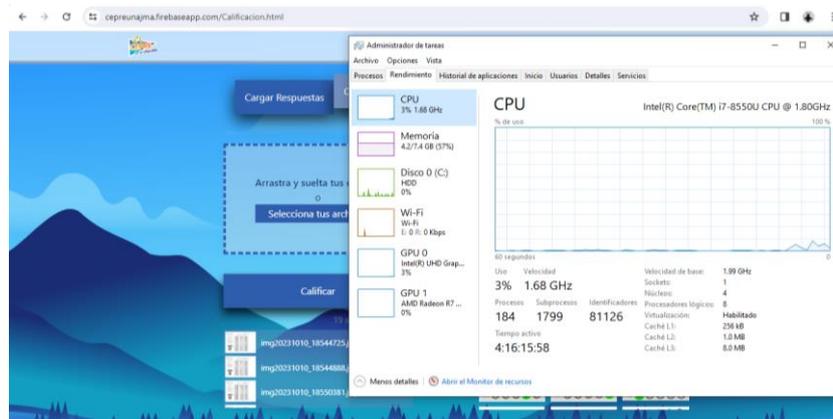


Figura 57

Pruebas de rendimiento en la calificación de 30 cartillas.

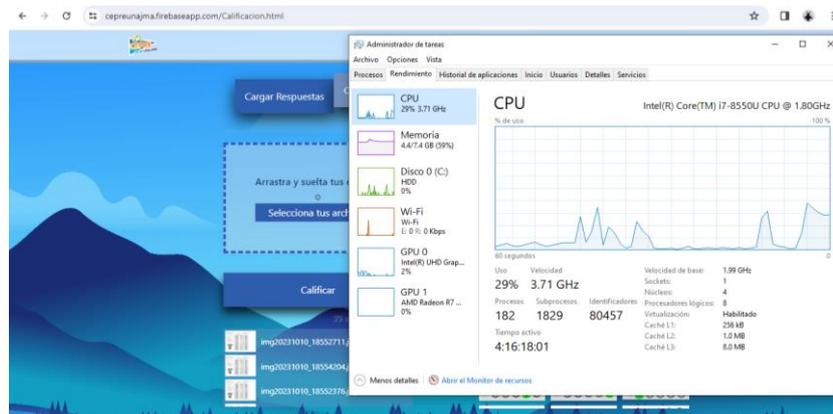


Figura 58

Pruebas de rendimiento en la calificación de 50 cartillas.

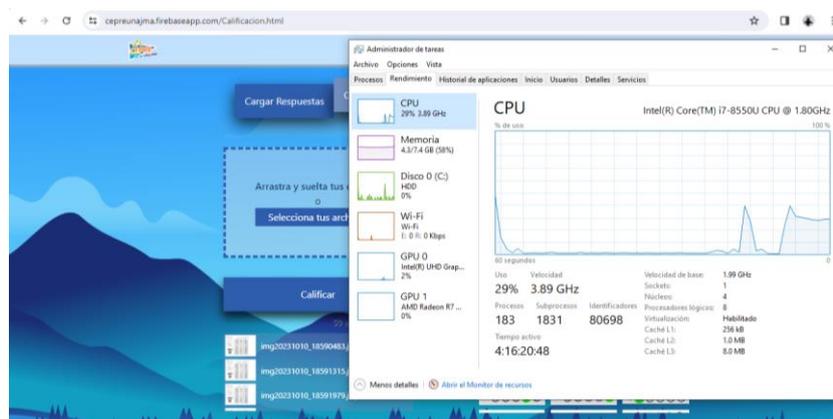


Figura 59

Pruebas de rendimiento en la calificación de 75 cartillas.

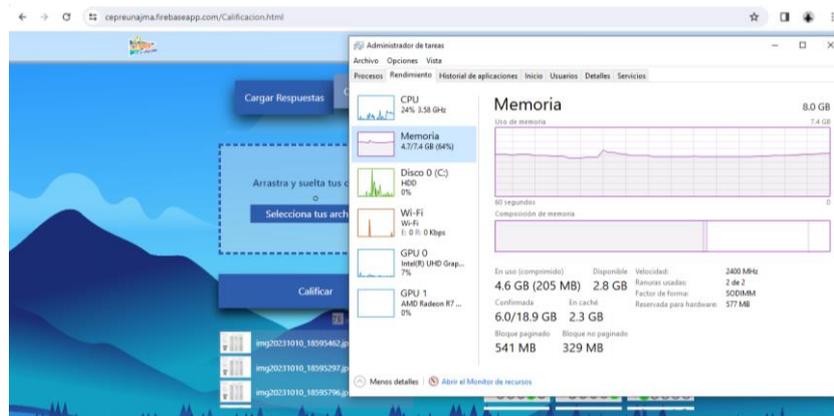
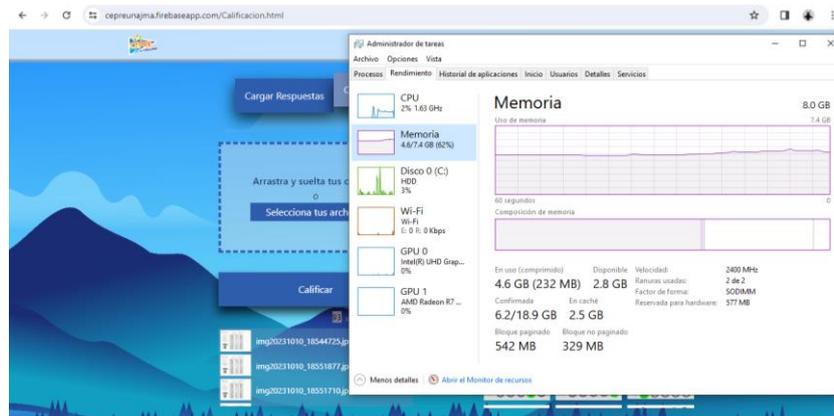


Figura 60

Pruebas de rendimiento en la calificación de 93 cartillas.



PRUEBAS DE USABILIDAD

Figura 61

Pruebas de usabilidad de registro de estudiantes - editar.

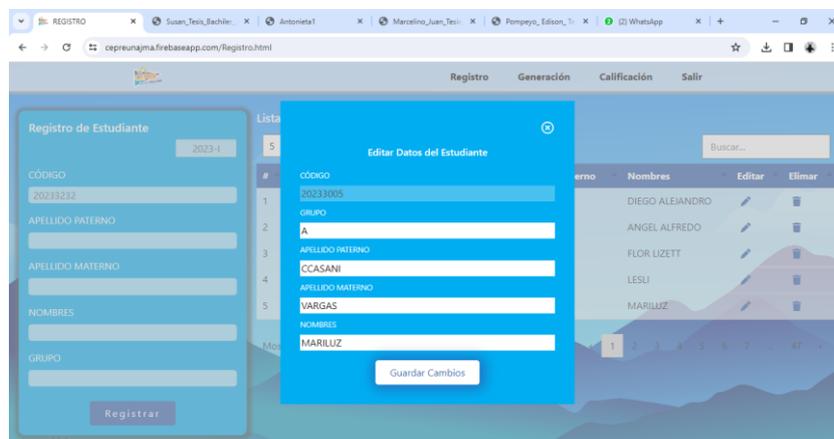


Figura 62

Pruebas de usabilidad: estructura, rendimiento.

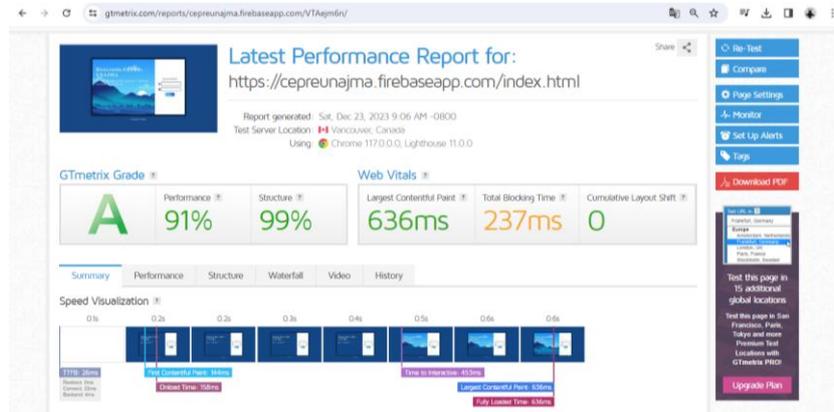
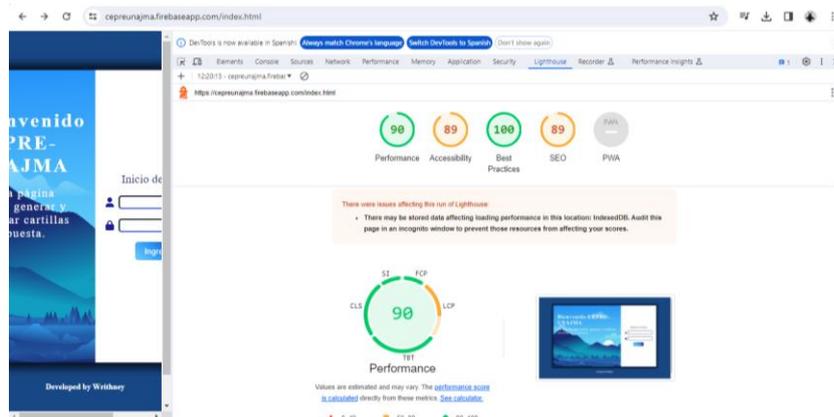


Figura 63

Pruebas de usabilidad: rendimiento, accesibilidad, buenas prácticas, CEO.



PRUEBAS DE COMPATIBILIDAD

Figura 64

Pruebas de portabilidad del sistema, versión web.

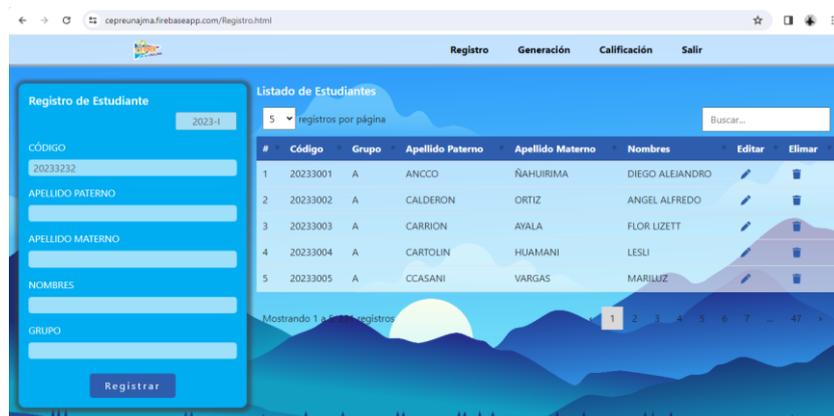


Figura 65

Pruebas de portabilidad del sistema, versión Mobile.

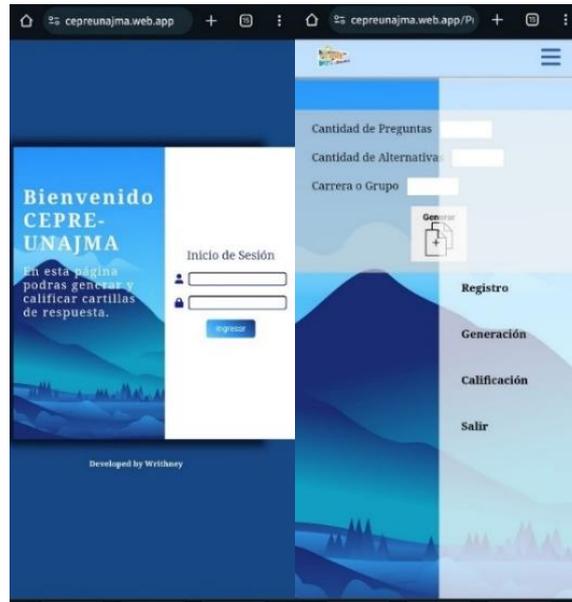
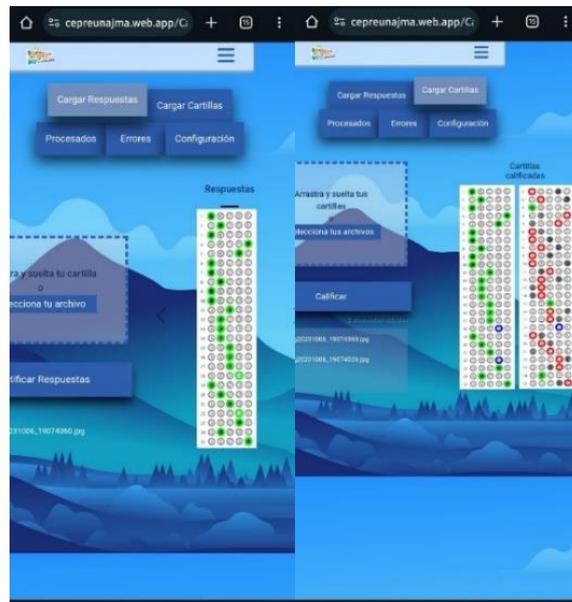


Figura 66

Pruebas de portabilidad del sistema, calificación versión Mobile.



PRUEBAS DE SEGURIDAD

Figura 67

Pruebas seguridad con herramienta 1.

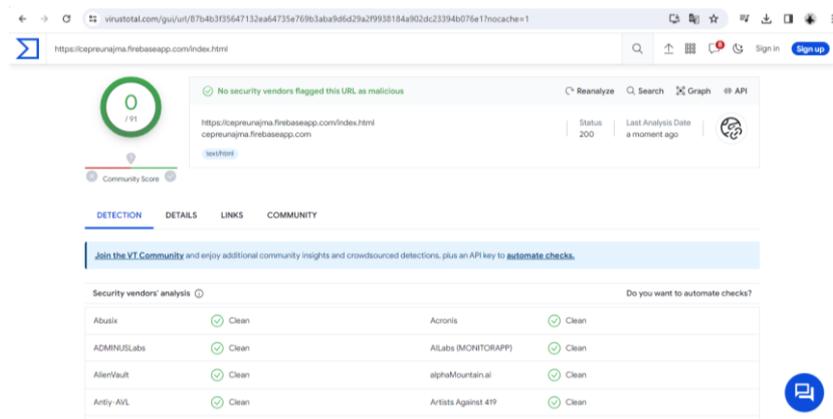


Figura 68

Pruebas seguridad con herramienta 2.

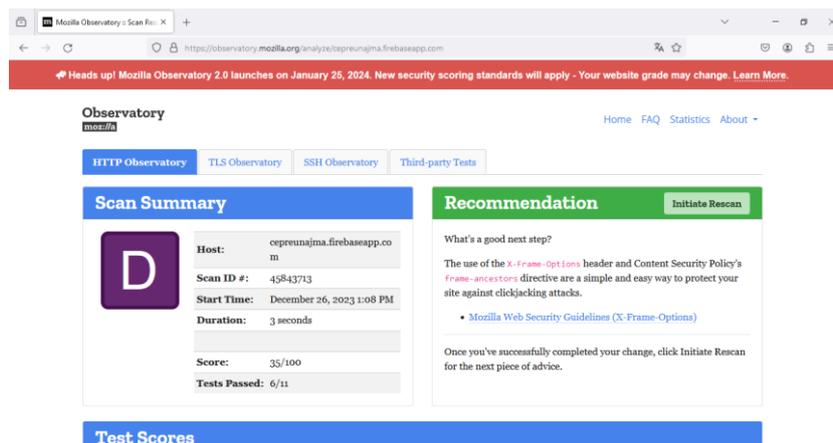


Figura 69

Pruebas seguridad con herramienta 3.

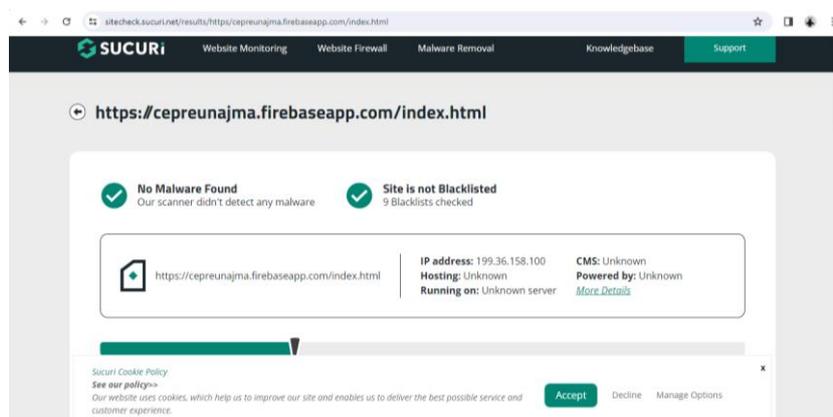
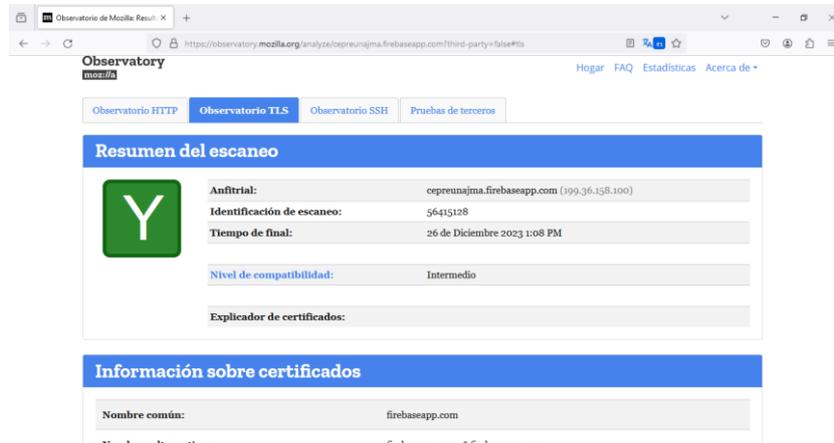


Figura 70

Pruebas seguridad con herramienta 4.



PRUEBAS UNITARIAS

Figura 71

Prueba unitaria de calificación de cartillas.

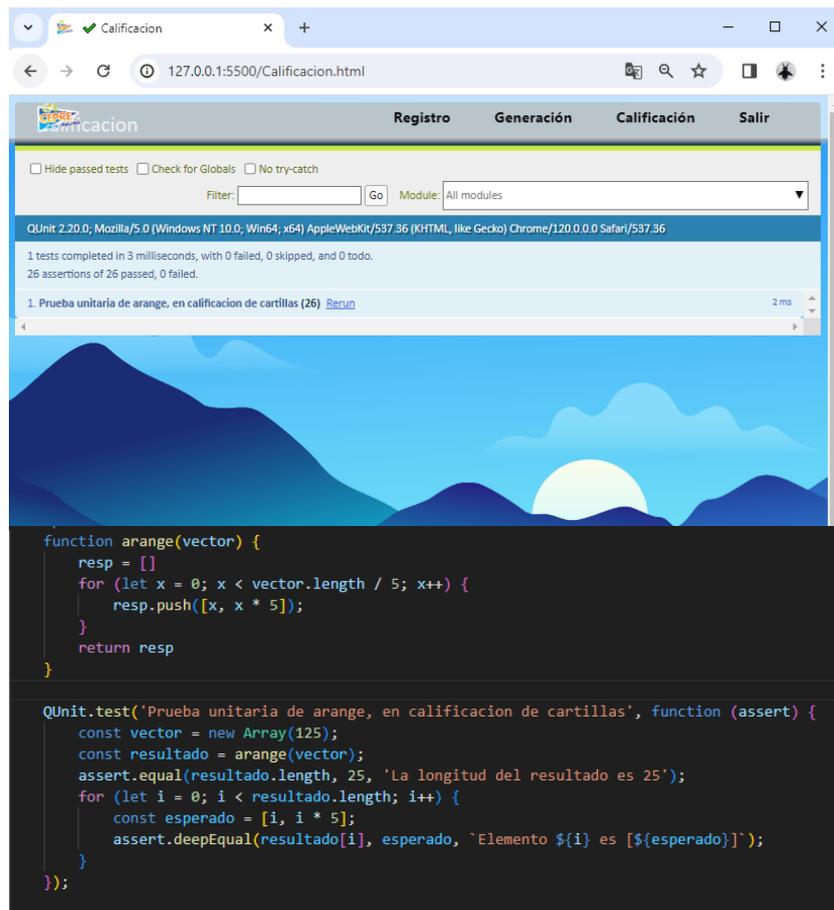
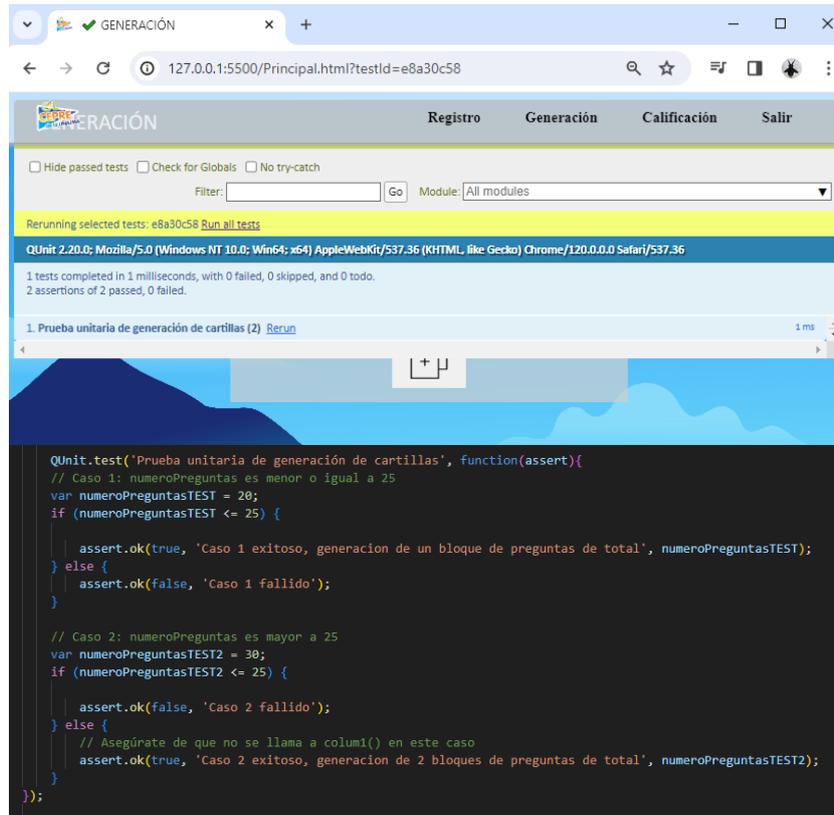


Figura 72

Prueba unitaria de generación de cartillas.



PRUEBAS DE INTEGRACIÓN

Figura 73

Pruebas de integración, envío de configuración de parámetros para procesar las cartillas.

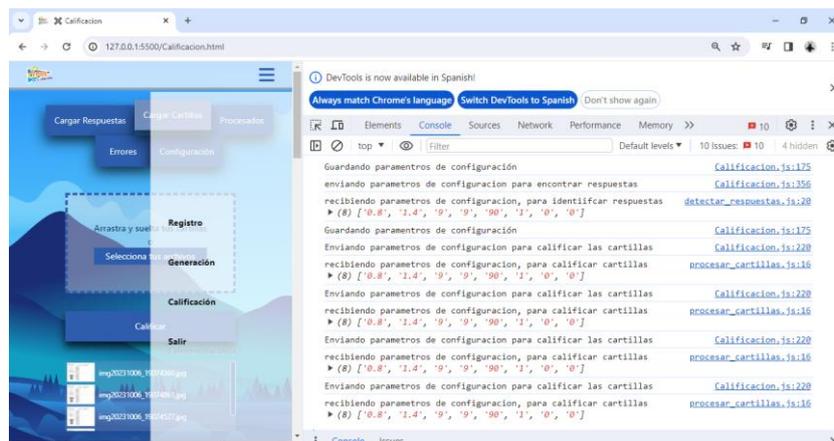


Figura 74

Pruebas de integración, inicio de sesión y autenticación con Firebase.

```
function login() {  
  let user = document.getElementById("email").value;  
  let pass = document.getElementById("password").value;  
  if (user == "" || pass == "") {  
    console.log('Faltan Ingresar datos');  
    alert('Faltan ingresar datos');  
  } else {  
    firebase.auth().signInWithEmailAndPassword(user, pass)  
      .then((userCredential) => {  
        var user = userCredential.user;  
        console.log('Comprobando credenciales, credenciales correctas');  
      })  
      .catch((error) => {  
        alert('Credenciales incorrectas');  
        console.log('Usuario o contraseña incorrecto');  
        password.value = '';  
        email.value = '';  
        email.focus()  
      });  
  }  
}  
document.getElementById("btn-login").addEventListener("click", login);
```

**ANEXO 02 FICHA DE RECOPIACIÓN PARA OBJETIVO ESPECIFICO 01 (GRUPO1 DE
25 PREGUNTAS)**

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

**FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE
SISTEMAS**

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE
CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO
DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

DENOMINACION DEL INSTRUMENTO: FICHA DE RECOPIACIÓN

En el siguiente cuadro se recopila los tiempos de calificación por cartilla del GRUPO1 DE 25 PREGUNTAS en las configuraciones de (800x566px, 1600x1132px, 3506x2481px), en 19 cartillas de respuesta; los tiempos de calificación por cartilla están en milisegundos.

# de cartilla	Dimensiones de la cartilla		
	800 x 566px	1600 x 1132px	3506 x 2481px
	Tiempos de calificación por cartilla en milisegundos		
1	65	238	1123
2	84	237	1123
3	63	228	1095
4	77	246	1115
5	65	230	1117
6	65	232	1088
7	64	260	1075
8	65	224	1120
9	72	266	1123
10	71	240	1080
11	68	225	1137
12	71	226	1145
13	64	234	1105
14	65	263	1110
15	81	234	1133
16	73	225	1134
17	93	261	1127
18	86	232	1118
19	68	281	1122
Promedio	71.57894737	241.1578947	1115.263158

**ANEXO 03 FICHA DE RECOPIACIÓN PARA OBJETIVO ESPECIFICO 01 (GRUPO2 DE
50 PREGUNTAS)**

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

**FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE
SISTEMAS**

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE
CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO
DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

DENOMINACION DEL INSTRUMENTO: FICHA DE RECOPIACIÓN

En el siguiente cuadro se recopila los tiempos de calificación por cartilla del GRUPO2 DE 50 PREGUNTAS en las configuraciones de (800x566px, 1600x1132px, 3506x2481px), en 60 cartillas de respuesta; los tiempos de calificación por cartilla están en milisegundos.

# de cartilla	Dimensiones de la cartilla		
	800 x 566px	1600 x 1132px	3506 x 2481px
	Tiempos de calificación por cartilla en milisegundos		
1	120	934	2086
2	144	552	1765
3	118	456	1832
4	120	493	1763
5	117	441	1144
6	103	500	1880
7	101	480	1726
8	123	394	1916
9	105	439	1747
10	100	401	1712
11	109	526	2118
12	129	426	2127
13	100	449	1757
14	103	411	1751
15	98	490	1765
16	99	393	1812
17	99	452	2164
18	99	309	1412
19	100	417	2044
20	97	377	1819
21	98	446	2951
22	101	427	1868
23	111	477	1746
24	110	456	1802
25	98	419	1933
26	102	452	1819
27	100	405	1733
28	104	449	1777
29	127	435	1916

30	115	391	1732
31	108	407	2033
32	122	424	1812
33	98	422	1972
34	98	522	1764
35	105	433	1757
36	108	431	1778
37	100	453	1739
38	110	469	1913
39	110	402	1932
40	98	412	1749
41	103	463	1777
42	103	457	1767
43	114	424	1904
44	103	552	1706
45	96	454	2054
46	110	354	2104
47	105	459	1963
48	101	393	1780
49	118	430	1770
50	99	1879	1852
51	105	509	1763
52	97	406	2286
53	98	455	1923
54	103	441	2193
55	156	1785	2156
56	106	519	1817
57	98	414	1797
58	102	481	1717
59	104	375	1878
60	129	451	1838
Promedio	107.6166667	496.2166667	1868.516667

ANEXO 04 FICHA DE RECOPIACIÓN PARA OBJETIVO ESPECIFICO 01 (GRUPO3 DE 75

PREGUNTAS)

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

DENOMINACION DEL INSTRUMENTO: FICHA DE RECOPIACIÓN

En el siguiente cuadro se recopila los tiempos de calificación por cartilla del GRUPO3 DE 75 PREGUNTAS en las configuraciones de (800x566px, 1600x1132px, 3506x2481px), en 75 cartillas de respuesta; los tiempos de calificación por cartilla están en milisegundos.

# de cartilla	Dimensiones de la cartilla		
	800 x 566px	1600 x 1132px	3506 x 2481px
	Tiempos de calificación por cartilla en milisegundos		
1	160	827	2913
2	145	562	1908
3	153	566	2434
4	137	523	2469
5	136	547	1756
6	148	581	2230
7	133	557	2247
8	130	542	2157
9	104	540	2162
10	162	593	2158
11	159	537	2883
12	134	509	2286
13	140	522	2243
14	138	482	2501
15	133	480	2249
16	133	503	2211
17	135	677	2185
18	177	716	2223
19	176	553	2434
20	136	527	2387
21	133	608	2280
22	135	568	2206
23	143	545	2287
24	132	499	2650
25	141	514	2249
26	136	516	2261
27	133	534	2230
28	150	577	2304
29	133	574	1705
30	138	642	2262
31	147	568	2410
32	136	512	2312
33	131	511	2318
34	138	567	2384
35	138	512	2198
36	133	499	2355

37	133	1252	2212
38	135	530	2280
39	132	486	2203
40	131	548	2295
41	133	710	2335
42	160	827	2913
43	132	509	2347
44	145	592	2296
45	168	592	2259
46	135	755	2217
47	148	511	2260
48	129	537	2339
49	131	917	2940
50	135	566	2392
51	131	490	2283
52	162	698	2377
53	145	571	2226
54	132	556	2201
55	165	573	2159
56	143	550	2200
57	138	639	2219
58	135	591	2259
59	134	548	2196
60	280	545	2197
61	209	687	2351
62	373	1179	2543
63	178	496	2332
64	134	485	2338
65	132	594	2391
66	164	854	2305
67	133	700	2209
68	220	718	2247
69	205	450	2202
70	987	632	2243
71	147	638	2250
72	569	582	2370
73	177	708	2342
74	400	603	2239
75	1318	542	2250
Promedio	185.2933333	597	2292.946667

ANEXO 05 FICHA DE RECOPIACIÓN PARA OBJETIVO ESPECIFICO 02

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

DENOMINACION DEL INSTRUMENTO: FICHA DE RECOPIACIÓN

En la tabla a continuación, se recopilan un total de 45,375 marcas correspondientes al 80% de las cartillas totales (153). Se ha utilizado la configuración óptima de 800x566 píxeles. Las dimensiones presentadas están expresadas en píxeles. Dada la extensa cantidad de datos, se adjunta un enlace a una tabla en Excel alojada en la nube:

Link	Capturas																																																																																																										
https://docs.google.com/spreadsheets/d/1bdFGQ6JZXRpHKv2JYCDDq8HQ80EGHXvM/edit?usp=drive_link&ouid=104057793017979717502&rtpof=true&sd=true	 <table border="1" data-bbox="1040 1176 1305 1518"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> </tr> <tr> <th>1</th> <th>Ancho</th> <th>Alto</th> <th>Proporción</th> <th></th> </tr> </thead> <tbody> <tr><td>2</td><td>13</td><td>12</td><td>1.0833333</td><td></td></tr> <tr><td>3</td><td>13</td><td>12</td><td>1.0833333</td><td></td></tr> <tr><td>4</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>5</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>6</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>7</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>8</td><td>13</td><td>12</td><td>1.0833333</td><td></td></tr> <tr><td>9</td><td>12</td><td>12</td><td>1</td><td></td></tr> <tr><td>10</td><td>14</td><td>13</td><td>1.0769231</td><td></td></tr> <tr><td>11</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>12</td><td>13</td><td>12</td><td>1.0833333</td><td></td></tr> <tr><td>13</td><td>12</td><td>12</td><td>1</td><td></td></tr> <tr><td>14</td><td>14</td><td>12</td><td>1.1666667</td><td></td></tr> <tr><td>15</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>16</td><td>13</td><td>13</td><td>1</td><td></td></tr> <tr><td>17</td><td>13</td><td>12</td><td>1.0833333</td><td></td></tr> <tr><td>18</td><td>12</td><td>12</td><td>1</td><td></td></tr> <tr><td>19</td><td>12</td><td>13</td><td>1.0730769</td><td></td></tr> </tbody> </table> <table border="1" data-bbox="555 1550 917 1675"> <tbody> <tr> <td>Total de marcas analizadas</td> <td>45375</td> </tr> <tr> <td>Maximo Valor de marca</td> <td>1.3333333</td> </tr> <tr> <td>Minimo Valor de marca</td> <td>0.8</td> </tr> </tbody> </table>		A	B	C	D	1	Ancho	Alto	Proporción		2	13	12	1.0833333		3	13	12	1.0833333		4	13	13	1		5	13	13	1		6	13	13	1		7	13	13	1		8	13	12	1.0833333		9	12	12	1		10	14	13	1.0769231		11	13	13	1		12	13	12	1.0833333		13	12	12	1		14	14	12	1.1666667		15	13	13	1		16	13	13	1		17	13	12	1.0833333		18	12	12	1		19	12	13	1.0730769		Total de marcas analizadas	45375	Maximo Valor de marca	1.3333333	Minimo Valor de marca	0.8
	A	B	C	D																																																																																																							
1	Ancho	Alto	Proporción																																																																																																								
2	13	12	1.0833333																																																																																																								
3	13	12	1.0833333																																																																																																								
4	13	13	1																																																																																																								
5	13	13	1																																																																																																								
6	13	13	1																																																																																																								
7	13	13	1																																																																																																								
8	13	12	1.0833333																																																																																																								
9	12	12	1																																																																																																								
10	14	13	1.0769231																																																																																																								
11	13	13	1																																																																																																								
12	13	12	1.0833333																																																																																																								
13	12	12	1																																																																																																								
14	14	12	1.1666667																																																																																																								
15	13	13	1																																																																																																								
16	13	13	1																																																																																																								
17	13	12	1.0833333																																																																																																								
18	12	12	1																																																																																																								
19	12	13	1.0730769																																																																																																								
Total de marcas analizadas	45375																																																																																																										
Maximo Valor de marca	1.3333333																																																																																																										
Minimo Valor de marca	0.8																																																																																																										

ANEXO 06 FICHA DE RECOPIACIÓN PARA OBJETIVO ESPECIFICO 03

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE

SISTEMAS

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

DENOMINACION DEL INSTRUMENTO: FICHA DE RECOPIACIÓN

En la tabla a continuación, se recopilan un total de 36535 umbrales en blanco y 8842 umbrales marcados en el 80% de las cartillas totales (153), usando la configuración optima de 800x566px. Dada la extensa cantidad de datos, se adjunta un enlace a una tabla en Excel alojada en la nube:

Link	Capturas												
https://docs.google.com/spreadsheets/d/1dclWh90pWhgg7hcd0RTQYpL7dexqIptL/e/dit?usp=drive_link&ouid=104057793017979717502&rt_pof=true&sd=true	 <table border="1" data-bbox="555 1487 1369 1563"> <thead> <tr> <th></th> <th></th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>Total de Umbrales Blancos Analizados</td> <td>36534</td> <td>Total de Umbrales Marcados Analizados</td> <td>8840</td> </tr> <tr> <td>Valor maximo de Umbral para blancos</td> <td>88</td> <td>Valor minimo de marcados</td> <td>91</td> </tr> </tbody> </table>					Total de Umbrales Blancos Analizados	36534	Total de Umbrales Marcados Analizados	8840	Valor maximo de Umbral para blancos	88	Valor minimo de marcados	91
Total de Umbrales Blancos Analizados	36534	Total de Umbrales Marcados Analizados	8840										
Valor maximo de Umbral para blancos	88	Valor minimo de marcados	91										

ANEXO 07 METODOLOGÍA DEL DESARROLLO DEL SISTEMA

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

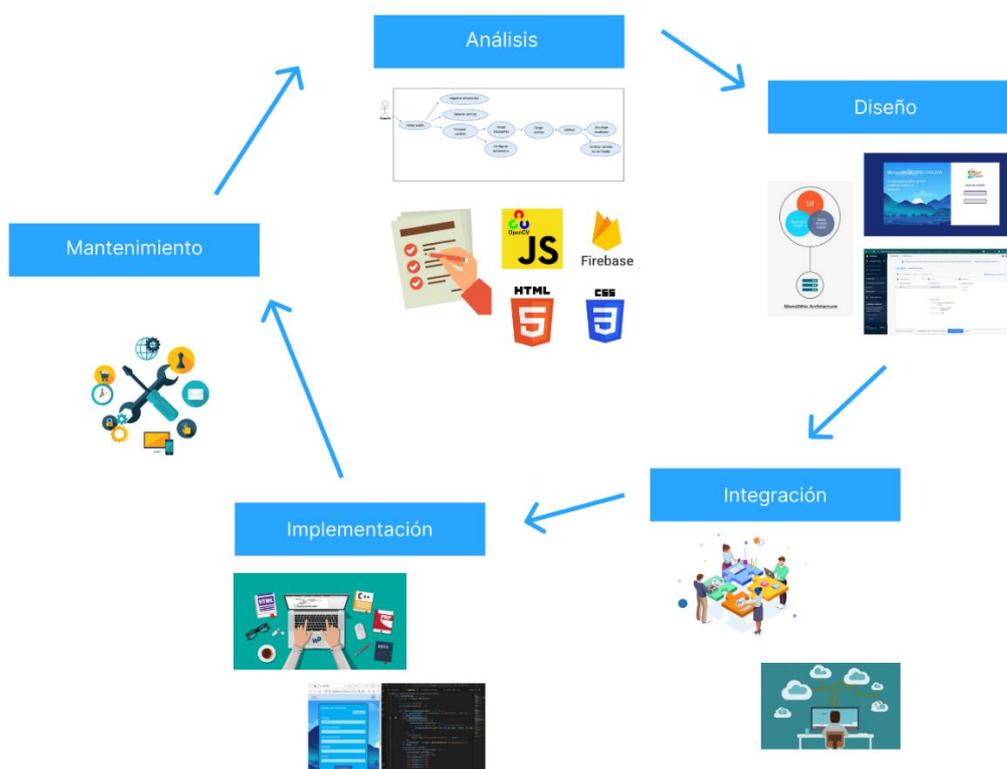
FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE
SISTEMAS

TÍTULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE
CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO
DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

Figura 75

Metodología del desarrollo del sistema.



ANEXO 08 DESARROLLO DEL SISTEMA

UNIVERSIDAD NACIONAL JOSÉ MARÍA ARGUEDAS

FACULTAD DE INGENIERÍA - ESCUELA PROFESIONAL DE INGENIERÍA DE

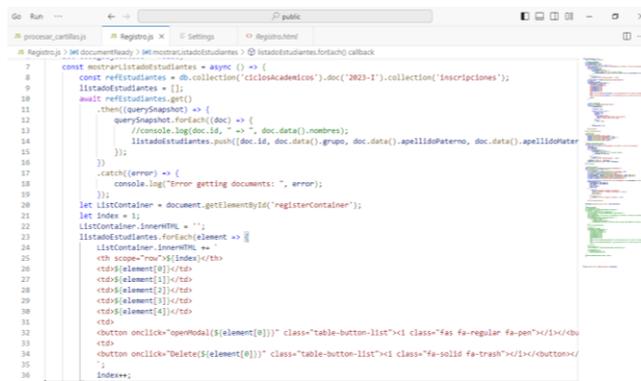
SISTEMAS

TITULO DEL PROYECTO DE TESIS: SISTEMA PARA LA GENERACIÓN Y CALIFICACIÓN DE CARTILLAS DE EXÁMENES DE SELECCIÓN MÚLTIPLE BASADO EN RECONOCIMIENTO ÓPTICO DE MARCAS Y VISIÓN COMPUTACIONAL

NOMBRE Y APELLIDOS DEL AUTOR: WRITHNEY XIMENA LAZO MARTÍNEZ

Figura 76

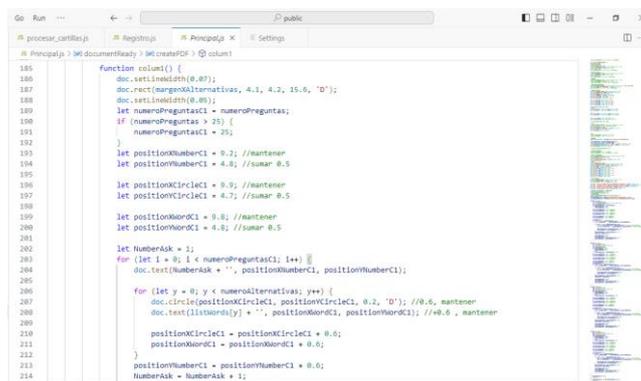
Módulo de registro de estudiantes - mostrar todos los registros.



```
7  # procesar_cartillas.js  # Registros  # Settings  # Registros.html
8  # Registros.js | document.readyState | document.readyState | @ listaEstudiantes.forEach callback
9  const mostrarListaEstudiantes = async () => {
10   const refEstudiantes = db.collection('ciclosacademicos').doc('2023-I').collection('inscripciones');
11   listaEstudiantes = [];
12   await refEstudiantes.get()
13   .then(querySnapshot => {
14     querySnapshot.forEach(doc => {
15       //console.log(doc.id, ' => ', doc.data().nombres);
16       listaEstudiantes.push({doc.id, doc.data().grupo, doc.data().apellidoPaterno, doc.data().apellidoMater
17     });
18   })
19   .catch(error => {
20     console.log("Error getting documents: ", error);
21   });
22   let listContainer = document.getElementById('registerContainer');
23   let index = 1;
24   listContainer.innerHTML = '';
25   listaEstudiantes.forEach(element => {
26     listContainer.innerHTML += `
27     <tr>
28       <td>${element[0]}</td>
29       <td>${element[1]}</td>
30       <td>${element[2]}</td>
31       <td>${element[3]}</td>
32     </tr>
33     <button onclick="openModal(${element[0]})" class="table-button-list"><i class="fas fa-regular fa-pen"></i></button>
34     <button onclick="Delete(${element[0]})" class="table-button-list"><i class="fa-solid fa-trash"></i></button>
35     `;
36   }
37   index++;
38 }
```

Figura 77

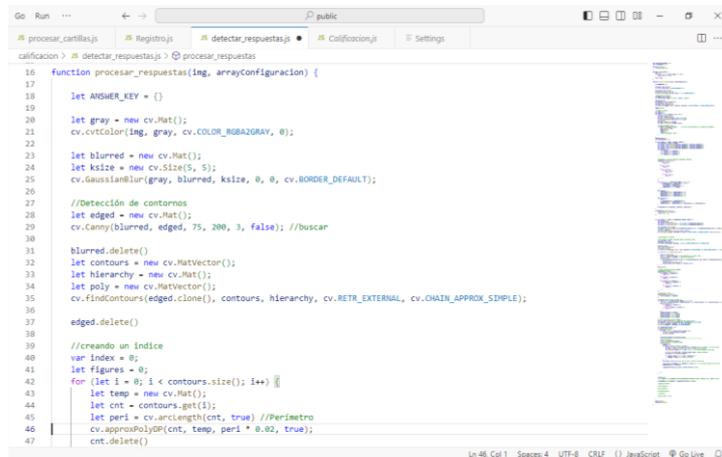
Generación de cartillas, primer bloque de preguntas.



```
185  # procesar_cartillas.js  # Registros  # Principios.js  # Settings
186  # Principios.js | document.readyState | document.readyState | @ column
187  function column() {
188   doc.setLineWidth(0.87);
189   doc.rect(margenAlternativas, 4.1, 4.2, 15.6, 'D');
190   doc.setLineWidth(0.85);
191   let numeroPreguntas = numeroPreguntas;
192   if (numeroPreguntas > 25) {
193     numeroPreguntas = 25;
194   }
195   let positionNumberC1 = 0.2; //mantener
196   let positionNumberC1 = 4.9; //sumar 0.5
197   let positionCircleC1 = 0.9; //mantener
198   let positionCircleC1 = 4.7; //sumar 0.5
199   let positionMorC1 = 9.8; //mantener
200   let positionMorC1 = 4.8; //sumar 0.5
201
202   let NumberAsk = 1;
203   for (let i = 0; i < numeroPreguntas; i++) {
204     doc.text(NumberAsk, positionNumberC1, positionNumberC1);
205
206     for (let y = 0; y < numeroAlternativas; y++) {
207       doc.circle(positionCircleC1, positionCircleC1, 0.2, 'D'); //0.6, mantener
208       doc.text(listMorC1[y], positionMorC1, positionMorC1); //0.6, mantener
209
210       positionCircleC1 = positionCircleC1 + 0.6;
211       positionMorC1 = positionMorC1 + 0.6;
212     }
213     positionNumberC1 = positionNumberC1 + 0.6;
214     NumberAsk = NumberAsk + 1;
215   }
216 }
```

Figura 78

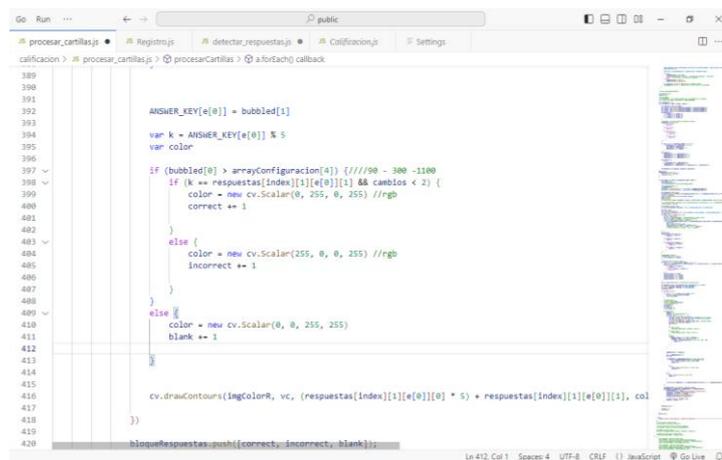
Reconocimiento de contornos de respuestas correctas.



```
16 function procesar_respuestas(img, arrayConfiguracion) {
17     let ANSWER_KEY = []
18
19     let gray = new cv.Mat();
20     cv.cvtColor(img, gray, cv.COLOR_RGBA2GRAY, 0);
21
22     let blurred = new cv.Mat();
23     let ksize = new cv.Size(5, 5);
24     cv.GaussianBlur(gray, blurred, ksize, 0, 0, cv.BORDER_DEFAULT);
25
26     //Detección de contornos
27     let edged = new cv.Mat();
28     cv.Canny(blurred, edged, 75, 200, 3, false); //buscar
29
30     blurred.delete();
31     let contours = new cv.MatVector();
32     let hierarchy = new cv.Mat();
33     let poly = new cv.MatVector();
34     cv.findContours(edged.clone(), contours, hierarchy, cv.RETR_EXTERNAL, cv.CHAIN_APPROX_SIMPLE);
35
36     edged.delete();
37
38     //creando un indice
39     var index = 0;
40     let figures = 0;
41     for (let i = 0; i < contours.size(); i++) {
42         let temp = new cv.Mat();
43         let cnt = contours.get(i);
44         let peri = cv.arcLength(cnt, true) //Perimetro
45         cv.approxPolyDP(cnt, temp, peri * 0.02, true);
46         cnt.delete();
47     }
```

Figura 79

Clasificación de respuestas correctas, incorrectas y en blanco.



```
389
390
391     ANSWER_KEY[e[0]] = bubbled[1]
392
393     var k = ANSWER_KEY[e[0]] % 5
394     var color
395
396
397     if (bubbled[0] > arrayConfiguracion[4]) { //90 - 300 - 1100
398         if (k == respuestas[index][1][e[0]][1] && cambios < 2) {
399             color = new cv.Scalar(0, 255, 0, 255) //rgb
400             correct += 1
401         }
402     }
403     else {
404         color = new cv.Scalar(255, 0, 0, 255) //rgb
405         incorrect += 1
406     }
407 }
408
409 else {
410     color = new cv.Scalar(0, 0, 255, 255)
411     blank += 1
412 }
413
414
415     cv.drawContours(imgColorR, vc, (respuestas[index][1][e[0]][0] * 5) + respuestas[index][1][e[0]][1], col
416
417 ))
418
419
420     bloqueRespuestas.push([correct, incorrect, blank]);
```