

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

**SISTEMA WEB PARA ANÁLISIS GRAFOLÓGICO BASADO EN
MACHINE LEARNING**

**DESARROLLO DE UN SISTEMA WEB BASADO EN GRAFOLOGÍA
Y MACHINE LEARNING PARA RECOMENDACIÓN
DE PAREJAS AFINES SENTIMENTALMENTE.**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO
PARA LA OBTENCIÓN EL TÍTULO DE INGENIERO EN SOFTWARE**

MARCELA ALEJANDRA MONTALVO LEÓN

marcela.montalvo@epn.edu.ec

DIRECTOR: PhD. MARTÍNEZ MOSQUERA SILVIA DIANA

diana.martinez@epn.edu.ec

DMQ, agosto 2024

CERTIFICACIONES

Yo, MARCELA ALEJANDRA MONTALVO LEON declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A handwritten signature in blue ink that reads "Marcela Montalvo". The signature is written in a cursive style with a horizontal line underlining the last part of the name.

MARCELA MONTALVO

Certifico que el presente trabajo de integración curricular fue desarrollado por MARCELA ALEJANDRA MONTALVO LEÓN bajo mi supervisión.

PhD. MARTÍNEZ MOSQUERA SILVIA DIANA
DIRECTORA

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

MARCELA ALEJANDRA MONTALVO LEÓN

KEVIN MAURICIO RIVADENEIRA BEJARANO

RONALD STALIN CARGUA VILANA

PhD. MARTÍNEZ MOSQUERA SILVIA DIANA

DEDICATORIA

Este trabajo de titulación es resultado del apoyo incondicional de mi madre, mi pilar fundamental en mi carrera universitaria. Por lo que dedico el presente trabajo a ella y a mi familia por su respaldo en sus diferentes formas, a mi hijo Sam por ser mi inspiración y motivación, y a mi padre por inculcarme el sueño de convertirme en una gran ingeniera. Este logro es un reflejo del amor, la confianza y el apoyo que he recibido de todos ustedes, a quienes dedico con profundo agradecimiento este trabajo.

AGRADECIMIENTO

Al concluir esta etapa esencial de mi formación profesional, deseo expresar mi sincero agradecimiento a quienes han sido fundamentales en mi camino como ingeniera.

Primero, agradezco a mis profesores por su sabiduría y dedicación, en especial a la Ing. Diana Martínez, mi tutora de TIC, por su guía y apoyo en mi investigación y desarrollo. También valoro el apoyo constante de mi colega y amigo Ing. Marco Logacho, quien me ha asistido en cada aspecto de mi carrera. Agradezco a María Fernanda Heredia, grafóloga, por su valiosa contribución única a este proyecto.

Mención especial para mi compañero de carrera Kevin Rivadeneria, con quien he compartido numerosas tareas y proyectos, convirtiéndonos en grandes amigos. Agradezco su amistad y apoyo. A mi prima y confidente, Eva Montalvo, por su incondicional apoyo en todo momento.

Finalmente, agradezco a todos mis compañeros de carrera por su amistad y compañerismo. Juntos hemos vivido experiencias inolvidables y aprendido mucho.

Dedico este trabajo a todos ustedes con profunda gratitud.

ÍNDICE DE CONTENIDO

1	INTRODUCCIÓN.....	1
1.1	Objetivo general.....	1
1.2	Objetivos específicos.....	1
1.3	Alcance.....	2
1.4	Marco teórico.....	2
1.4.1	Grafología como método para analizar la personalidad mediante la escritura.....	2
1.4.2	Teoría de los cuatro humores.....	3
1.4.3	Metodología CRISP-DM.....	4
1.4.4	Metodología Kanban en el Desarrollo de Software.....	5
1.4.5	Machine Learning.....	7
	Machine Learning Supervisado.....	7
1.4.6	Aprendizaje Profundo.....	7
	Las redes neuronales convolucionales.....	7
1.4.7	Aprendizaje por Transferencia.....	8
1.4.8	Desarrollo Web.....	9
	Front-end.....	9
	Back-end.....	10
	Microservicios.....	11
	Redis.....	11
1.4.9	Jira para proyectos Kanban.....	11
1.4.10	Confluence para base de conocimiento.....	11
2	METODOLOGÍA.....	12
2.1	Análisis Inicial.....	12
2.1.1	Análisis del negocio.....	12
2.1.2	Valoración del contexto.....	13
2.2	Metodología CRISP-DM.....	14
2.2.1	Fase 1: Entendimiento del negocio.....	15
	Objetivos del negocio.....	15
	Objetivos Específicos.....	15
	Valoración de la situación.....	15
	a. Alcance de plan de proyecto.....	23
	b. Descripción de tareas por fases.....	23
2.2.2	Fase 2: Comprensión de los datos.....	25
	Recolección de datos.....	25
	Descripción de los datos.....	26
	Exploración de los datos.....	27
	Verificación de la calidad de los datos.....	28
	Preparación de los datos.....	29
2.2.3	Fase 3: Selección de datos.....	29

Limpieza de Datos.....	29
Estructura de Datos.....	30
Integración de Datos.....	31
Formateo de Datos.....	31
2.2.4 Fase 4: Modelado.....	32
Selección de técnicas de modelado.....	32
Diseño de pruebas.....	33
Construcción del Modelo.....	34
e. Data Augmentation.....	35
f. Parámetros para división de conjuntos.....	35
g. Parámetros del modelo.....	35
Evaluación del modelo.....	36
2.2.5 Fase 5: Fase de Despliegue.....	37
Plan de despliegue.....	37
Monitoreo.....	38
2.2.6 Fase 6: Informe final.....	38
Revisión del Proyecto:.....	39
2.3 Metodología Kanban en proyectos de Software.....	39
2.3.1 Cálculo del Work in Progress (WIP).....	41
Adaptación del tablero Kanban.....	41
Métricas de rendimiento.....	42
Historias de usuario HU.....	42
Tareas.....	42
Flujo de trabajo.....	43
Análisis y diseño.....	45
Diagrama de base de datos.....	45
Arquitectura de microservicios.....	47
Prototipos de interfaz de usuario.....	48
Ejecución.....	48
Testing y Despliegue.....	49
3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	51
3.1 Resultados.....	51
Metodología CRISP-DM.....	51
API Publicada.....	54
Metodología Kanban.....	55
FronD-End conectada a Back end.....	55
Autenticación con Firebase.....	58
4 Conclusiones y Recomendaciones.....	59
4.1 Conclusiones.....	59
4.2 Recomendaciones.....	60
5 Referencia bibliográfica.....	62
6 ANEXOS.....	64
ANEXO I.....	65
ANEXO II.....	71
ANEXO III.....	73

RESUMEN

El proyecto desarrolla un sistema web innovador que integra un modelo de *machine learning* para clasificar firmas manuscritas en cuatro tipos de personalidad: bilioso, nervioso, linfático y sanguíneo. Utilizando la metodología CRISP-DM para la creación del modelo y Kanban para el desarrollo del software, el proyecto aborda la falta de profesionales en grafología en Ecuador, proporcionando una herramienta que permite la evaluación de personalidades y la recomendación de personas compatibles a nivel psicológico. El modelo de *machine learning* se entrenó con la técnica de *transfer learning* usando *Inception V3*, lo que permitió aprovechar un modelo preentrenado y optimizar recursos. Las evaluaciones del modelo mostraron mejoras constantes en precisión y pérdida, y las métricas indicaron un nivel aceptable de clasificación, aunque con áreas de mejora en las clases de bilioso y linfático. La implementación de Kanban facilitó una gestión ágil y eficiente del proyecto, mejorando la productividad del equipo. El sistema web resultante permite a los usuarios cargar imágenes de sus firmas y obtener un análisis detallado de su personalidad, así como identificar personas compatibles para relaciones duraderas y satisfactorias. El despliegue de la API y la integración del modelo en la infraestructura en la nube aseguran la disponibilidad y escalabilidad del sistema.

PALABRAS CLAVE: Grafología, Machine Learning, Transfer Learning, Inception V3, CRISP-DM, Kanban.

ABSTRACT

The project develops an innovative web system that integrates a machine learning model to classify handwritten signatures into four personality types: bilious, nervous, lymphatic, and sanguine. Using the CRISP-DM methodology for model creation and Kanban for software development, the project addresses the lack of graphology professionals in Ecuador by providing a tool that allows personality evaluations and the recommendation of psychologically compatible individuals. The machine learning model was trained using the transfer learning technique with Inception V3, leveraging a pre-trained model to optimize resources. Model evaluations showed consistent improvements in accuracy and loss, and metrics indicated an acceptable level of classification, though with areas for improvement in the bilious and lymphatic classes. The implementation of Kanban facilitated agile and efficient project management, enhancing team productivity. The resulting web system allows users to upload images of their signatures and obtain a detailed personality analysis, as well as identify compatible individuals for long-lasting and satisfying relationships. The API deployment and model integration into the cloud infrastructure ensure the system's availability and scalability.

KEYWORDS: Graphology, Machine Learning, Transfer Learning, Inception V3, CRISP-DM, Kanban.

1 INTRODUCCIÓN

Este proyecto sugiere la creación de un sistema web de análisis grafológico utilizando técnicas de Machine Learning para determinar los rasgos psicológicos de una persona. Una vez identificado el perfil psicológico, el sistema indicará una personalidad afín sentimentalmente según su personalidad.

La grafología se define como la evaluación de los rasgos físicos y los patrones dentro de la escritura a mano de un individuo para comprender su estado psicológico al momento de la escritura. La escritura a mano es una prueba proyectiva en la que el inconsciente se manifiesta en lo consciente.

Existen teorías sobre la construcción de la personalidad y métodos de emparejamiento según el área de aplicación, muchas de las cuales se basan en cinco dimensiones: sociabilidad, empatía, diligencia, equilibrio emocional y receptividad a nuevas experiencias.

Este sistema ayudará a superar las limitaciones de subjetividad presentes en el emparejamiento manual. Con un variado conjunto de datos de escritura, se entrenará un modelo de aprendizaje automático para reconocer patrones y correlaciones imperceptibles para el ojo humano. A medida que el sistema se perfeccione, podrá proporcionar recomendaciones más precisas y útiles.

1.1 Objetivo general

Desarrollar un sistema web basado en grafología y machine learning para recomendación de parejas afines sentimentalmente en base al análisis de rasgos y patrones de la firma manuscrita.

1.2 Objetivos específicos

- Realizar una revisión de la literatura académica y científica relacionada con la grafología orientada hacia temperamentos y técnicas de *machine learning*.
- Abstracter información de estudios previos y metodologías implementadas en sistemas similares para contrastar el tipo de tecnología empleada.
- Crear un *dataset* etiquetado a través de la recolección de firmas manuscritas para entrenar, probar y validar el modelo desarrollado.
- Desarrollar una solución web interactiva que permita la carga de muestras de escritura, extracción y análisis de rasgos psicológicos y recomendación y recomendación de parejas compatibles.

- Implementar pruebas del sistema para evaluar el cumplimiento de funcionalidades.

1.3 Alcance

Este proyecto se enfoca en la creación de una plataforma web que involucra análisis grafológico y *machine learning*, con el propósito de proporcionar recomendaciones precisas y personalizadas a los usuarios en diversos ámbitos. El sistema podrá recomendar parejas afines sentimentalmente.

El desarrollo de este sistema implica un conjunto de actividades bien definidas. Inicialmente, se llevará a cabo un análisis detallado de la literatura científica y técnica en el área de la grafología, el análisis de personalidad y las recomendaciones sentimentales. Seguidamente, se construirá una base de datos diversa y representativa de escrituras manuscritas, para entrenar al modelo de *machine learning* en el reconocimiento de rasgos de personalidad.

Posteriormente, se desarrollará el modelo de *machine learning*, que se basará en algoritmos de visión por computadora para extraer atributos grafológicos clave de las imágenes de escritura cargadas por los usuarios. La validación y ajuste riguroso del modelo garantizarán su eficacia en la identificación precisa de rasgos de personalidad. En paralelo, se diseñará y evaluará la interfaz del sistema web, asegurando su facilidad de uso.

Finalmente, se implementará un algoritmo de recomendación basado en la personalidad, que comparará los perfiles de los usuarios y generará recomendaciones de parejas afines. Estas recomendaciones se presentarán de manera intuitiva, resaltando las compatibilidades detectadas.

1.4 Marco teórico

En el marco teórico presente se detallan los fundamentos teóricos en los que se encuentra basado el proyecto. Esto incluye conceptos del negocio como la grafología guiada a los temperamentos hipocráticos, las metodologías utilizadas para llevar el proyecto constituido por proyectos de datos con algoritmos de *machine learning* y desarrollo de aplicaciones web con tecnologías específicas para Back-End y Front-End.

1.4.1 Grafología como método para analizar la personalidad mediante la escritura

El método de grafología se basa en la idea de que la escritura refleja directamente el cerebro humano, permitiendo revelar aspectos del carácter, emociones y comportamientos. A pesar de su popularidad, la grafología enfrenta escepticismo en la comunidad científica, ya que la American Psychological Association (APA) no la reconoce como una ciencia establecida debido a la falta de pruebas empíricas consistentes [1]. Sin embargo, algunos grafólogos afirman que características como la presión del trazo, la inclinación de las letras y el tamaño de la escritura pueden ofrecer *insights* sobre la personalidad, aunque estas interpretaciones no están respaldadas por estudios científicos robustos [2].

Los grafólogos creen que ciertos aspectos de la escritura pueden indicar rasgos específicos de la personalidad. Por ejemplo, la escritura inclinada hacia la derecha podría sugerir una persona extrovertida, mientras que la escritura pequeña y apretada podría señalar una persona detallista y reservada [3]. Además, existen herramientas y software que intentan automatizar el análisis grafológico mediante algoritmos como *Grafometrics*, aunque su eficacia y precisión son objeto de debate y no sustituyen la interpretación de un grafólogo profesional [4].

1.4.2 Teoría de los cuatro humores

Hipócrates propuso que el cuerpo de las personas está compuesto por cuatro humores: sangre, flema, bilis amarilla y bilis negra, asociados a los cuatro elementos básicos del universo: tierra, agua, fuego y aire [5]. La salud dependía del equilibrio y la adecuada temperatura de estos humores. Cada persona tenía una disposición única de estos humores, con uno predominando sobre los demás. Este equilibrio también variaba según las estaciones del año [5].

En la *Figura 1*, se presenta un resumen de los cuatro temperamentos con sus respectivas características:

Elemento	Cualidades	Humor	Tipo	Estación
AIRE	Caliente- <u>Humedo</u>	Sangre	<u>Sanguinario/ Sanguíneo</u>	Primavera
FUEGO	Caliente - Seco	Bilis Amarilla	<u>Colérico/ Bilioso</u>	Verano
TIERRA	Fría - Seca	Bilis Negra	<u>Melancólico/ Nervioso</u>	Otoño
AGUA	Fría - Húmeda	Flema	<u>Flemático/ Linfático</u>	Invierno

Figura 1 Temperamentos según Hipócrates [5]

Cada tipo de temperamento se encuentra compuesta por las cualidades y humor de los elementos que usa Hipócrates para su clasificación.

Con el fin de filtrar únicamente la información necesaria para este trabajo se toman los adjetivos calificativos de personalidad de cada tipo [5]:

✓ Sanguíneo

Físico: Robusto, bien desarrollado, frente amplia, labios carnosos, mirada expresiva.

Carácter: Dinámico, optimista, sociable, carismático, líder natural.

Social: Extrovertido, comunicativo, empático, buen oyente, pero puede ser despótico e invasivo.

Grafología: Tamaño: Grande, Forma: Curva, ampulosa, Presión: Firme, Velocidad: Rápida, Inclinación: Hacia la derecha, Dirección: Ascendente

! Nervioso

Físico: Delgado, rostro afilado y pálido, mirada vivaz.

Carácter: Intuitivo, creativo, emotivo, ambivalente (extrovertido e introvertido), versátil.

Social: Variable según el entorno, difícilmente encajable, impulsivo.

Grafología: Tamaño: Normal, Forma: Variable, filiforme, Presión: Irregular, Velocidad: Rápida, Inclinación: Oscilante, Dirección: Flexible, variable

⊖ Bilioso

Físico: Esbelto, rostro anguloso, mirada profunda.

Carácter: Reflexivo, analítico, realista, sobrio, serio, introvertido.

Social: Distante, firme en decisiones, controlado emocionalmente, poco expresivo.

Grafología: Tamaño: Pequeño, Forma: Sobria, angular, Presión: Firme, Velocidad: Mesurada o rápida, Inclinación: Vertical o ligeramente inclinada, Dirección: Recta, rígida

⚠ Linfático

Físico: Obesidad, movimientos lentos, pasivo.

Carácter: Analítico, metódico, regular, adaptable, conformista.

Social: Tranquilo, introvertido, no expresivo, convencional.

Grafología: Tamaño: Normal, pequeño o grande, Forma: Curva, redondeada, Presión: Pastosa, Velocidad: Lenta, Inclinación: Variable (recta, inclinada o invertida), Dirección: Descendente o recta

Figura 2 Característica por tipo de temperamento

La Figura 2 describe las características físicas, de carácter, sociales y grafológicas de los temperamentos sanguíneo, bilioso, nervioso y linfático. El sanguíneo es robusto y extrovertido; el bilioso es esbelto y analítico; el nervioso es delgado e intuitivo; y el linfático tiende a la obesidad y es metódico. Socialmente, el sanguíneo es comunicativo pero invasivo, el bilioso es firme pero poco expresivo, el nervioso es versátil pero difícilmente encajable, y el linfático es tranquilo y convencional. Grafológicamente, el sanguíneo presenta escritura grande y curva, el bilioso pequeña y sobria, el nervioso normal y variable, y el linfático puede tener formas curvas y redondeadas.

1.4.3 Metodología CRISP-DM

El Proceso Estándar de Minería de Datos (Cross-Industry Standard Process for Data Mining) usado para diversas industrias es un enfoque comúnmente empleado en proyectos de análisis de datos. Proporciona un proceso estructurado y bien definido que ayuda a los equipos a desarrollar soluciones de análisis de datos eficientemente [6]. Consta de seis fases principales [7]:

1. **Comprensión del Negocio:** Establecimiento de metas y necesidades.
2. **Comprensión de los Datos:** Recolección, análisis y valoración de la calidad de los datos.
3. **Preparación de los Datos:** Procesamiento, selección e integración de datos.
4. **Modelado:** Elección y aplicación de métodos de modelado.
5. **Evaluación:** Validación de los modelos para asegurar que cumplan con los objetivos.
6. **Despliegue:** Implementación del modelo en el entorno de producción.

CRISP-DM se representa a través de un modelo jerárquico que organiza las tareas en distintos niveles de abstracción, incluyendo Fase, Tarea Genérica y Tarea Especializada. Lo que permite organizar el trabajo en tareas asignadas a una fase con entregables esperados en cada tarea que compondrían los entregables de cada fase [8]

Así, cada fase está dividida en tareas que contienen a tareas más específicas que guían toda la metodología, el flujo se describe en **Figura 3**.

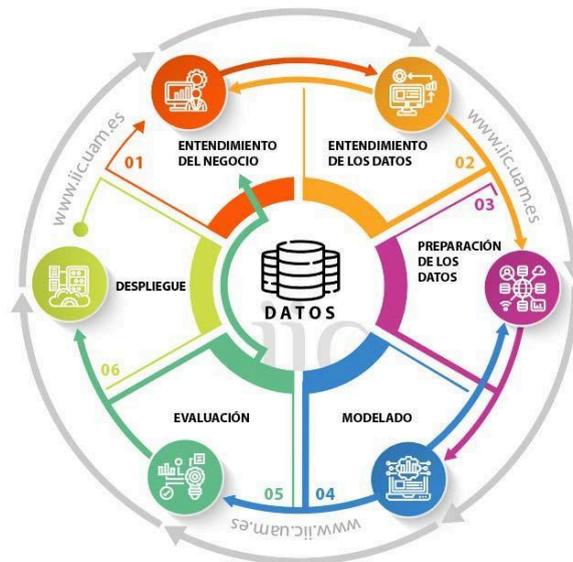


Figura 3 Esquema del ciclo CRISP-DM estándar [7]

El modelo CRISP-DM es flexible y permite personalizaciones según las necesidades del proyecto, sin seguir una secuencia estricta, lo que facilita su adaptación a distintos contextos. Ampliamente utilizado en diversas industrias, ofrece un marco robusto y adaptable que asegura la calidad en minería de datos [6], [9].

1.4.4 Metodología Kanban en el Desarrollo de Software

Kanban es una metodología ágil y visual utilizado en la industria de software enfocado en la entrega "justo a tiempo" (JIT) y la gestión eficiente del trabajo en progreso (WIP). En lugar de iteraciones, Kanban promueve la evolución continua del proceso. Utiliza tarjetas, físicas o virtuales, que representan tareas en el desarrollo de software (SDLC) [10].

Principios Fundamentales:

- **Visualización del Trabajo:** Los tableros Kanban muestran el flujo de trabajo y las etapas, como "Pendiente", "En Progreso" y "Terminado". Esto mejora la comunicación y la colaboración, especialmente en equipos distribuidos.
- **Limitación del Trabajo en Progreso (WIP):** Se establecen límites para evitar la sobrecarga de trabajo y mejorar el enfoque y la productividad.
- **Gestión del Flujo:** Se optimiza el flujo de trabajo para asegurar una entrega continua y eficiente.
- **Mejora Continua:** Kanban promueve una cultura de mejora constante, utilizando métricas para evaluar y mejorar la eficiencia del equipo.

Para implementar Kanban, se debe comenzar con un tablero en blanco y crear columnas que representen las etapas del trabajo. Las tareas se añaden al tablero y avanzan a través de las columnas hasta completarse. Es importante limitar el número de tareas en progreso y establecer políticas claras para guiar el flujo de trabajo [11].

Cuando aplicamos Kanban al desarrollo de software no se encuentran definidos roles específicos como en otros frameworks ágiles [12] se puede considerar los roles de Gerente de Entrega de Servicios (SDM)/Gerente de Flujo y Gerente de Solicitudes de Servicio (SRM) que no están ligados a ninguna norma rígida de definición ni responsabilidades y se pueden adaptar al contexto del proyecto y equipo disponible.

Aunque las pizarras de Kanban cuentan con tres columnas predefinidas de “Por Hacer”, “En progreso” y “Completado”, al adaptar a un SDLC pueden llegar a ser tan complejas como el flujo de trabajo lo defina [12], según el ejemplo de *Microsoft* con *Azure Boards*, una pizarra de Kanban con las fases de desarrollo puede verse como en la Figura 4.

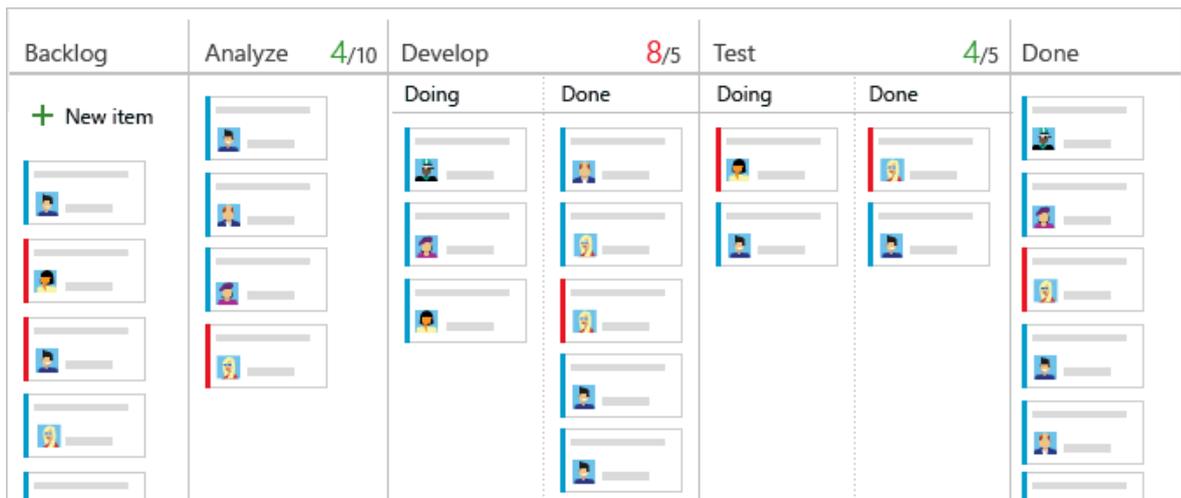


Figura 4 Tablero Kanban adaptado a SLDC [13]

1.4.5 Machine Learning

El aprendizaje automático (ML), una subdisciplina de la inteligencia artificial (IA), usa datos y algoritmos para replicar y mejorar el aprendizaje humano. Su proceso se basa en tres elementos clave: realizar predicciones o clasificaciones, evaluar su precisión mediante una función de error y ajustar el modelo para optimizar su exactitud. [14].

Machine Learning Supervisado

Usa datos etiquetados para entrenar algoritmos, como redes neuronales y regresión lineal, a clasificar datos o predecir resultados. Es útil para problemas del mundo real como la clasificación de correos no deseados[15].

1.4.6 Aprendizaje Profundo

El aprendizaje profundo es una especialización del aprendizaje automático que utiliza redes neuronales con múltiples capas para identificar patrones complejos en grandes conjuntos de datos. Inspiradas en el cerebro humano, estas redes aprenden de manera jerárquica, extrayendo características cada vez más abstractas a medida que se avanzan en las capas. [16].

Las redes neuronales convolucionales

Las Redes Neuronales Convolucionales (CNN) son un tipo de red neuronal profunda optimizada para analizar datos con estructura de cuadrícula, como las imágenes. Han sido clave en el progreso de la visión por computadora, destacándose en el reconocimiento de imágenes. [17].

Arquitectura [18]:

- **Capas Convolucionales:** Son el núcleo de las CNN. Con un filtro a los datos de entrada extraen detalles, como contornos, texturas y formas. La convolución combina dos conjuntos de información para identificar estas características.
- **Mapas de Características:** El resultado de aplicar un filtro sobre una entrada es un mapa de características que destaca ciertos aspectos de la entrada, como los bordes en una imagen.
- **Fully Connected Layers:** Son similares a las usadas en redes neuronales tradicionales y se encuentran al final de la red. Transforman las características extraídas por las capas convolucionales en las salidas finales, como las probabilidades de clase en un problema de clasificación.
- **Función de Activación:** Una de las funciones de activación más comunes en las CNN es la *ReLU (Rectified Linear Unit)*, que introduce no linealidad en la red al activar las neuronas solo cuando la entrada es positiva.

Entrenamiento de las CNN:

- **Forward Propagation:** Durante el paso de propagación hacia adelante, datos atraviesan las capas convolucionales, de *pooling* y completamente conectadas para producir la salida.
- **Backward Propagation y Actualización de Pesos:** La retropropagación se emplea para ajustar los pesos de los filtros y las capas totalmente conectadas, reduciendo la función de pérdida mediante algoritmos de optimización como el descenso de gradiente [19].

1.4.7 Aprendizaje por Transferencia

El aprendizaje por transferencia implica tomar un modelo preentrenado en un conjunto de datos y una tarea específicos, y ajustar (*fine-tuning*) o reutilizar partes de ese modelo para una nueva tarea. Esto se basa en que las características aprendidas por el modelo en la tarea original son útiles para la nueva tarea [20]

Los modelos preentrenados son aquellos que ya han sido entrenados en grandes conjuntos de datos, como ImageNet para imágenes o corpora extensos de texto para modelos de lenguaje. Existen dos tipos de aprendizaje por transferencia: el Fine-Tuning, que ajusta un modelo preentrenado entrenando algunas o todas sus capas adicionales con el nuevo conjunto de datos, y el Feature Extraction, que utiliza las características aprendidas por las capas del modelo preentrenado y entrena solo la capa de salida o un clasificador simple encima de esas características. Las fases del aprendizaje por transferencia incluyen [19]:

1. **Entrenamiento Inicial:** donde se entrena un modelo en un conjunto de datos grande y genérico.
2. **Transferencia:** Se emplea el modelo previamente entrenado y se adapta a la nueva tarea, lo que puede involucrar mantener invariables las capas iniciales del modelo y entrenar únicamente las capas finales., o ajustar todas las capas con una tasa de aprendizaje baja.
3. **Evaluación:** donde se valida el modelo en la nueva tarea y se ajustan los hiperparámetros según sea necesario.

1.4.8 Desarrollo Web

El desarrollo web consiste en crear y mantener sitios y aplicaciones web para navegadores, incluyendo el diseño de la interfaz y la experiencia del usuario, así como la implementación de funciones con lenguajes como HTML, CSS y JavaScript. Además,

abarca el uso de frameworks, administración de bases de datos y control del rendimiento. En este proyecto, implica integrar un sistema de machine learning para clasificar firmas manuscritas, asegurando una interfaz amigable y una funcionalidad adecuada que se adapte a las expectativas del usuario. [21]

Front-end

React es una librería de JavaScript de código abierto empleada para desarrollar interfaces de usuario, particularmente en aplicaciones web de página única. Desarrollada por Facebook, React se ha convertido en una de las herramientas más populares en el desarrollo front-end debido a su eficiencia, flexibilidad y enfoque basado en componentes[22].

Esta biblioteca puede ser utilizada en diferentes proyectos dependiente de su complejidad y recursos, además de integrar diversas herramientas [23].

El diagrama se divide en dos columnas. La columna izquierda, titulada 'Casos de uso', contiene tres puntos: 1. 'Aplicaciones de Una Sola Página (SPA)': React es ideal para aplicaciones de una sola página donde se requiere una experiencia de usuario dinámica y reactiva. 2. 'Aplicaciones Web Progresivas (PWA)': React puede ser utilizado para construir aplicaciones web progresivas que funcionan offline y ofrecen una experiencia similar a una aplicación nativa. 3. 'Aplicaciones de Alto Rendimiento': Gracias a su eficiente gestión del DOM y la arquitectura basada en componentes, React es adecuado para aplicaciones que requieren un alto rendimiento y una interfaz de usuario compleja. La columna derecha, titulada 'Herramientas', contiene tres puntos: 1. 'Create React App': Una herramienta oficial para crear proyectos de React con una configuración mínima. 2. 'Next.js': Un framework para React que permite el renderizado del lado del servidor y la generación de sitios estáticos. 3. 'Redux': Una biblioteca para la gestión del estado global de la aplicación. 4. 'React Native': Un framework para construir aplicaciones móviles nativas utilizando React.

Figura 5 React para desarrollo web

La Figura 5 muestra los usos y herramientas de *React* en el desarrollo web, destacando su aplicación en tres tipos de proyectos: aplicaciones de una sola página (SPA) para una experiencia de usuario dinámica, aplicaciones web progresivas (PWA) que funcionan offline y aplicaciones de alto rendimiento con una gestión eficiente del DOM. Además, se mencionan herramientas clave como *Create React App* para configuración mínima, *Next.js* para renderizado del lado del servidor, *Redux* para gestión del estado global y *React Native* para construir aplicaciones móviles nativas, subrayando la versatilidad de *React* en diversas necesidades de desarrollo web.

Back-end

NestJS es un *framework* progresivo para construir aplicaciones del lado del servidor con Node.js, basado en *TypeScript* y *JavaScript*. Sigue una arquitectura modular, proporcionando una estructura organizada y escalable que facilita la gestión y expansión de grandes proyectos. Los principios clave de *NestJS* incluyen el modularidad, que permite la separación y gestión de diferentes partes de la aplicación, y una arquitectura limpia que utiliza controladores, servicios y módulos para mantener una estructura ordenada. Los controladores gestionan las solicitudes HTTP, los servicios contienen la lógica empresarial y los módulos agrupan funcionalmente varias partes de la aplicación. Las ventajas de *NestJS* incluyen su escalabilidad, adecuado para aplicaciones de gran escala, facilidad de mantenimiento gracias a su arquitectura modular, y soporte para *TypeScript*, que aporta tipado fuerte y ventajas en el desarrollo [24]

Microservicios

Es una arquitectura de software que divide una aplicación en servicios independientes y de menor tamaño que se ejecutan de manera autónoma. Cada microservicio se centra en una funcionalidad específica y se comunica con otros servicios a través de APIs. Esto facilita la escalabilidad y el mantenimiento [25].

Redis

Es un repositorio de datos en memoria, de código abierto, usado como base de datos, caché y corredor de comunicaciones. Se destaca por su alta velocidad y soporte para formatos de datos como cadenas, hashes, entre otros. [26].

1.4.9 Jira para proyectos Kanban

Es un software de gestión de proyectos ampliamente aplicada, especialmente en entornos Agile. Su integración con Kanban proporciona un flujo de trabajo visual y eficiente para la gestión de tareas y proyectos. Con Jira Kanban, los equipos pueden visualizar el progreso de las tareas a través de tableros organizados en columnas (por ejemplo, Pendiente, En Progreso, y Completado). Esta visualización ayuda a identificar cuellos de botella y a gestionar el trabajo en progreso (WIP), permitiendo una entrega continua y mejorando la productividad [27]

1.4.10 Confluence para base de conocimiento

Confluence es una plataforma colaborativa que permite a los equipos crear, compartir y gestionar contenido. Actúa como una base de conocimiento centralizada donde se puede almacenar documentación, procedimientos, y cualquier tipo de información relevante para el proyecto. Esto es especialmente útil para equipos distribuidos. [28].

2 METODOLOGÍA

Para este proyecto se consideró la naturaleza de los componentes internos a desarrollar.

Es así como, para obtener un un modelo de aprendizaje automático que pueda de determinar la personalidad de un usuario mediante una imagen se utilizó la metodología CRISP-DM que es la metodología más común de minería de datos en varias industrias con una implementación flexible con enfoque en gestión de proyectos ágiles basado en equipo [29]

Mediante sus 6 fases de un proceso que se basa en tareas descritas en niveles de abstracción. Se tiene un modelo publicado para consumo del cliente por una Interfaz de Programación de Aplicaciones (API) para la comunicación entre diferentes softwares.

Para desarrollar el sistema web que integre al modelo y presente al usuario una interfaz de carga de imagen y visualización de resultados se utilizó la metodología Kanban adaptada a proyectos de software.

Se hizo un análisis para el entendimiento del negocio, previo al desarrollo de los componentes, para entender profundamente los objetivos y el alcance de cada componente.

2.1 Análisis Inicial

2.1.1 Análisis del negocio

El uso de la grafología para determinar el temperamento de una persona es una técnica poco utilizada en Ecuador debido a su alta complejidad y necesidad de un amplio conocimiento en análisis de escritura manuscrita y experiencia. Las empresas que requieren estos servicios contratan a empresas extranjeras para incluirlos en sus procesos forenses, determinación de fraudes [30] o en talento humano entre otros [31] . Mientras que los cursos en línea para aprender de esta técnica proyectiva han ido en aumento en plataformas como Udemy [32] o la Cámara de Comercio de Quito [33]

Lo que plantea que son pocos los profesionales que brindan este servicio como parte de un proceso de terapia individual o en pareja. Esto representa un problema para aquellas personas que buscan conocer a mayor detalle a una persona de forma complementaria.

Bajo este contexto, el negocio consiste en dar a conocer la personalidad de una persona a partir de su firma manuscrita generada de forma inconsciente, a través de un sistema web que determina la compatibilidad de dos usuarios a nivel psicológico, a través de la

integración de un modelo de aprendizaje automático capaz de etiquetar la imagen de una firma escrita según su temperamento.

2.1.2 Valoración del contexto

Cada persona, al tener una nueva interacción, controla, mide y actúa en consecuencia de cómo quiere ser percibida por los demás. Esta es una acción consciente que genera un interés o rechazo premeditado en el entorno social en que se produce. Para las interacciones efímeras sin interés romántico, esto no presenta un inconveniente mayor que cualquier posibilidad de desacuerdo. Para quienes buscan un vínculo a largo plazo, resulta complicado entablar una relación con alguien que, con el tiempo, no cumple las expectativas y genera una disonancia interna entre la persona proyectada y la que se conoce realmente.

Mientras esto ocurre, se pierde tiempo y oportunidades tratando de mantener un vínculo que, basado en la psicología de parejas, no tenía garantías de éxito.

Para generar confianza en el intento de conocer a una persona, se requiere una herramienta o análisis adicional que pueda ver más allá de lo que cada individuo quiere proyectar.

La grafología permite conocer el inconsciente de la persona, revelando su verdadero temperamento. Esta técnica de análisis puede ayudar a abordar el problema. Sin embargo, su efectividad depende de la capacidad, conocimiento y disponibilidad de los grafólogos en el país, así como del área de aplicación de esta técnica.

El objetivo de este proyecto es ofrecer el servicio de grafología a gran escala, permitiendo a cualquier persona conocer su personalidad y la compatibilidad con otra persona desde cualquier lugar. Y, se dirige a adultos mayores de 18 años que buscan conocer a otra persona en profundidad, para obtener un grado adicional de confianza antes de decidir entablar una conexión emocional.

El usuario debe ser consciente, en primer lugar, de sus propias características temperamentales, competencias, valores, atributos positivos y aspectos a mejorar. Además, espera que cada usuario del servicio pase por el mismo proceso de introspección, lo que genera confianza en el proceso. También desea que sus datos permanezcan privados y que el emparejamiento se realice con información generalizada y no sensible, incluyendo una descripción completa de su personalidad.

Las soluciones actuales para emparejar a personas bajo diferentes criterios como selección de Match con Scroll como lo realiza la plataforma de Tinder [34], incrementa la

problemática de conexiones superficiales y momentáneas que no satisfacen las expectativas de nuestro usuario objetivo. En contraposición Plenty of Fish [35] permite a los usuarios llenar un formulario sobre información personal y muestra sugerencia de usuarios con gustos similares, si bien es una mejora notable en el sistema de emparejamiento al garantizar un mínimo de intereses en común, este método aún se encuentra sujeto a manipulación o respuesta consciente del registro de las respuestas. Otra aplicación como Omegle [36] te presenta a la gente radicalmente a través de un chat o videocámara, lo que quita el problema de la intervención activa del usuario secundario, pero incurre en problemas de seguridad e integridad para el usuario.

Característica	Tinder	Plenty of Fish (POF)	Omegle	Bumble
Forma de Emparejamiento	Emparejamiento basado en deslizamientos, fotos y breves biografías.	Emparejamiento basado en perfiles detallados y cuestionarios de compatibilidad.	Emparejamiento basado en perfiles detallados y cuestionarios de compatibilidad.	Emparejamiento similar a Tinder con un giro: las mujeres deben iniciar la conversación después de un match.
Popularidad	MUY ALTA Una de las aplicaciones de citas más populares en todo el mundo.	ALTA Especialmente popular en América del Norte y otros países de habla inglesa.	MODERADO Conocida por el chat al azar, popular entre adolescentes y jóvenes.	ALTA Popular entre usuarios jóvenes y urbanos en busca de relaciones serias y conexiones.
Nivel de Éxito en Relaciones de Pareja	MODERADO Es conocida más por encuentros casuales que por relaciones serias, aunque hay casos de éxito.	RELATIVAMENTE ALTO Mayor enfoque en relaciones serias y matrimonios.	BAJO Principalmente utilizado para conversaciones casuales y conexiones temporales.	MODERADO Fomenta relaciones serias con el modelo de que las mujeres tomen la iniciativa, lo que puede conducir a conexiones más significativas.

Figura 6 Comparativa de aplicaciones similares

Por lo tanto, actualmente no existe un servicio de grafología enfocado a la comparación de personalidades con fines de vínculos emocionales.

Bajo este contexto se tiene que nuestro proyecto debe cumplir con los requerimientos del público objetivo fijado y se determinaron los siguientes criterios de éxito a cumplir:

2.2 Metodología CRISP-DM

Para controlar el flujo de tareas a realizar a lo largo de cada Fase de la metodología CRISP-DM se implementó un tablero Kanban. Al final de la Fase 1 con la construcción del plan de proyecto se pudo llenar las tareas dentro del tablero para su uso en las siguientes fases.

2.2.1 Fase 1: Entendimiento del negocio

Esta sección se busca tener una perspectiva del negocio para desarrollar el modelo de redes convolucionales y el proceso de obtención y tratamiento de datos.

Con los objetivos del negocio se obtiene un plan de proyecto que contempla objetivos, recursos, requerimientos y limitaciones para las siguientes fases y poder concretar objetivos de *data mining* congruentes con los objetivos primarios [8].

Objetivos del negocio

El objetivo del negocio es desarrollar un modelo de clasificación de personalidad a partir del análisis de imágenes de firmas manuscritas para mostrar personalidades compatibles con el usuario en términos de temperamento.

Objetivos Específicos

- Construir un conjunto de datos de imágenes de firmas manuscritas etiquetadas según los 4 temperamentos por un grafólogo experimentado, asegurando una base de conocimiento sólida.
- Desarrollar un modelo de clasificación que prediga la personalidad del usuario mediante el reconocimiento de imágenes.
- Determinar el modelo óptimo a implementar mediante la comparación y ajuste de modelos y técnicas aplicables al campo del negocio.
- Hacer disponibles los resultados de clasificación para su integración en procesos de emparejamiento según compatibilidad temperamental.

Valoración de la situación

Para esta sección se realizó una evaluación del contexto en que se desarrolla el proyecto para tomar decisiones informadas en las fases siguientes, obteniendo los siguientes resultados:

a. Inventario de Recursos

El inventario clasifica los recursos en dos categorías: esenciales y deseables, para facilitar su distinción. Cabe destacar que el conjunto de firmas fue planificado para ser recopilado, procesado y etiquetado con la autorización previa de los usuarios voluntarios que participaron en el proceso.



Figura 7 Inventario de recursos para data mining

La Figura 7 muestra el inventario de recursos necesarios para *data mining*, clasificándolos en esenciales y deseables. En el personal técnico esencial se incluyen un grafólogo etiquetador, un recolector de datos y un experto en *machine learning*. El *dataset* esencial consiste en un conjunto de 900 firmas manuscritas etiquetadas. En hardware, se considera esencial un ordenador de desarrollo con Windows 11, procesador Intel Core i7, 16 GB de RAM y 256 GB de almacenamiento, mientras que es deseable un servidor para despliegue con Windows Server 2022, procesador Intel Xeon, 12 GB de RAM y 128 GB de almacenamiento.

b. Criterios de Éxito

En los criterios de éxito dentro de la metodología, se toma como base los criterios de éxito obtenidos previamente en la sección 2.1.2, y se alinea específicamente con los resultados obtenidos por el modelo, logrando lo siguiente:

Criterios de éxito de Data Mining

En la siguiente lista especifican los criterios de éxito a cumplir por el data mining, se deben marcar como realizados al evaluar el modelo.

- Volumen de imágenes de firmas manuscritas almacenadas son de al menos 900 de las cuales se encuentren distribuidas uniformemente en los cuatro temperamentos de clasificación.
- Precisión de clasificación de firmas según las 4 categoría debe ser comparable con las nuevas etiquetas en tiempo real realizadas por el grafólogo.
- Desplegar el modelo con mejor rendimiento obtenido después de contrastar resultados obtenidos al finalizar las prácticas de transfer learning.

Figura 8 Checklist de criterios de éxito documentado en Confluence

La Figura 8 enumera los criterios de éxito para el proyecto de data mining, los cuales deben ser cumplidos al evaluar el modelo. Los criterios incluyen: tener al menos 900 imágenes de firmas manuscritas distribuidas uniformemente entre las cuatro categorías de temperamentos; lograr una precisión de clasificación de firmas comparable con las etiquetas en tiempo real realizadas por un grafólogo; y desplegar el modelo con el mejor rendimiento obtenido después de las prácticas de *transfer learning*.

c. Riesgos y Contingencias

Con el fin de estar preparados para cualquier escenario, se realizó una evaluación de riesgos mediante una matriz de evaluación de riesgo. Esta matriz sirvió de base para generar medidas de contingencia para cada riesgo identificado. La matriz de riesgos y las medidas de contingencia ilustrados en la **Figura 9**.

Matriz de Gestión de Riesgos

Riesgo	Descripción	Probabilidad	Impacto	Mitigación
Calidad Inadecuada de datos	La toma de datos puede no ser representativa de la población o tener un etiquetado erróneo.	Baja	Alto	Realizar una revisión global del etiquetado de firmas con el grafólogo experto, realizando comparaciones en tiempo real. Aplicar técnicas de validación cruzada para verificar la variabilidad de la muestra.
Fallo de Hardware o Software	Daño en los recursos informáticos disponibles para el desarrollo y despliegue del modelo	Baja	Alto	Manejar un controlador de versiones como GitHub para disponer del código en la nube. Realizar backups regulares del conjunto de datos en la nube o algún dispositivo de almacenamiento.
Retraso en el cronograma	Subestimación del tiempo necesario para la preparación del dataset, creación del modelo y entrenamiento	Media	Media	Desarrollar un seguimiento de tareas en el tiempo y acoplarlo regularmente para ajustarlo al desarrollo en tiempo real.

Figura 9 Matriz de gestión de riesgo y contingencia

d. Costos y Beneficios

Para determinar los costos y beneficios del modelo, se consideró la sección 2.1.2 que, debido a la naturaleza del proyecto de titulación, es necesario optimizar y reducir los costos al máximo. Además, se identifica al principal *stakeholder* como el grafólogo especializado que busca probar la automatización de su servicio dirigido al público objetivo. Con base en esta caracterización, se determinaron los siguientes costos y beneficios, detallados en la Figura 10 Costos y beneficios.

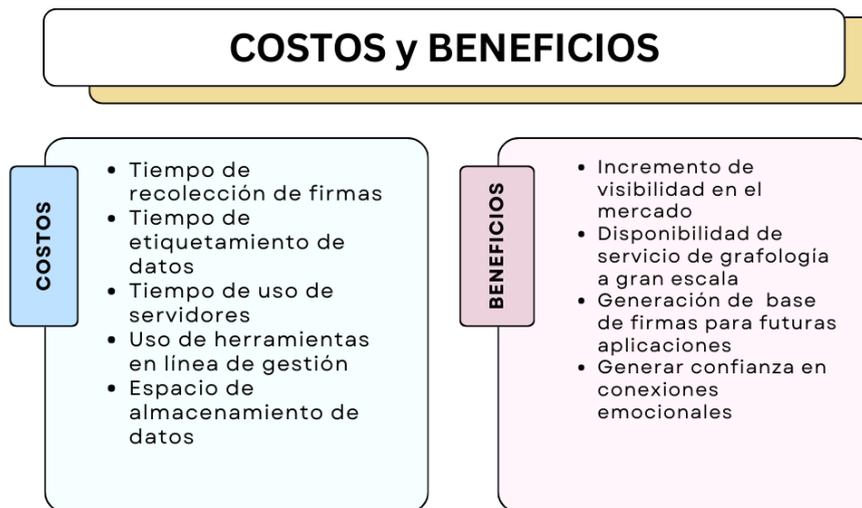


Figura 10 Costos y beneficios

Considerando tanto costos como beneficios propios del proyecto, se lo califica como **VIABLE**. Esto se debe a la colaboración entre la academia y el grafólogo, donde la academia aporta los insumos técnicos y el grafólogo contribuye con su tiempo y base de conocimientos.

e. Requerimientos

De acuerdo con los objetivos planteados y orientados a la lógica del negocio, los requerimientos del modelo se dividen en dos categorías: funcionales y no funcionales. Estos se describen en la Figura 11 y Figura 12.

Requisitos Funcionales

Toma de Datos	El modelo de machine learning debe aceptar y almacenar archivos en formato imagen de las firmas manuscritas
Procesamiento de firmas	La firma debe ser adecuada al requerimiento del modelo en cuanto a normalización, tamaño y formato
Etiquetado de datos	El dataset conformado debe ser etiquetado en base a las 4 personalidades: bilioso , linfático , nervioso y sanguíneo .
Clasificación	El modelo de machine learning debe asignar nuevas firmas a las 4 categorías de temperamento.
Generación de resultados	El modelo debe generar reportes de las métricas: <u>p</u> <u>precisión</u> , <u>p</u> <u>pérdida</u> , <u>f1-score</u> , <u>recall</u> y <u>matriz de confusión</u>
Almacenamiento	Se debe proveer de una base de datos segura para el almacenamiento y manejo de las firmas en base 64 y resultados de predicción.
Mejora del rendimiento	Se debe evaluar con variaciones de modelos preentrenados y parámetros dentro de las capas densas para encontrar el mayor rendimiento posible.

Figura 11 Requisitos funcionales

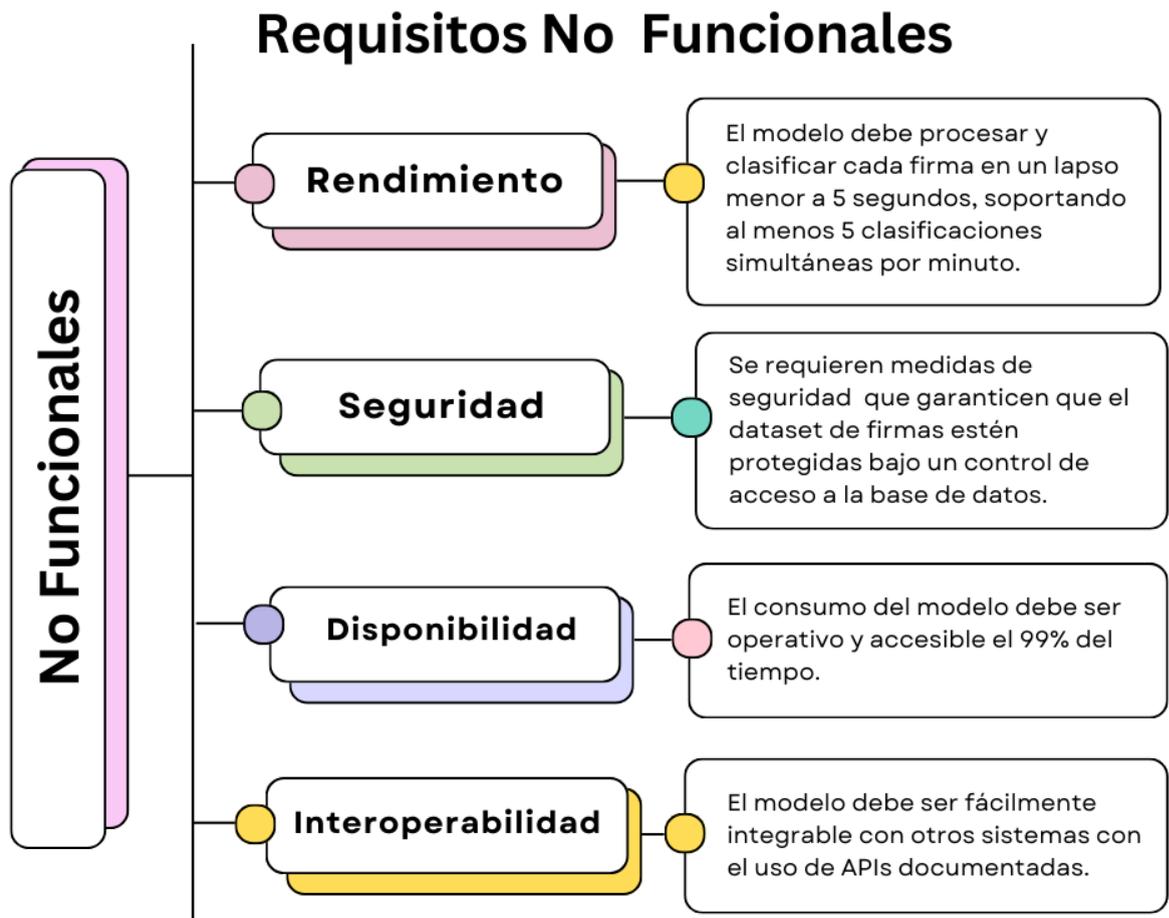


Figura 12 Requisitos no funcionales

El modelo se encuentra desplegado en la versión gratuita de Render, una plataforma de alojamiento web y despliegue de aplicaciones, la cual no brinda un servicio continuo debido a que está ligada a la concurrencia del consumo en esta versión. Por lo tanto, alcanzar un 99% de disponibilidad es un objetivo deseable dentro de los requisitos no funcionales.

f. Objetivos de Data Mining y Criterios de Éxito

A partir de las tareas previas, se establecen objetivos con una perspectiva más técnica, destacando las técnicas de *machine learning* y el flujo de datos.

Con base en estos objetivos, se determinaron los criterios de éxito que el modelo debe cumplir, clasificándolos según su posibilidad de ocurrencia:

- **Improbable:** El objetivo es difícil de alcanzar y enfrenta obstáculos significativos, requiriendo un esfuerzo considerable o recursos adicionales.
- **Posible:** El objetivo es alcanzable con un esfuerzo razonable y la aplicación de recursos adecuados. Existen desafíos, pero pueden superarse con una planificación y ejecución cuidadosas.
- **Probable:** Existe una alta probabilidad de lograr el objetivo. Los recursos necesarios están disponibles, los obstáculos son manejables y el camino hacia el éxito está relativamente claro.

Data Mining

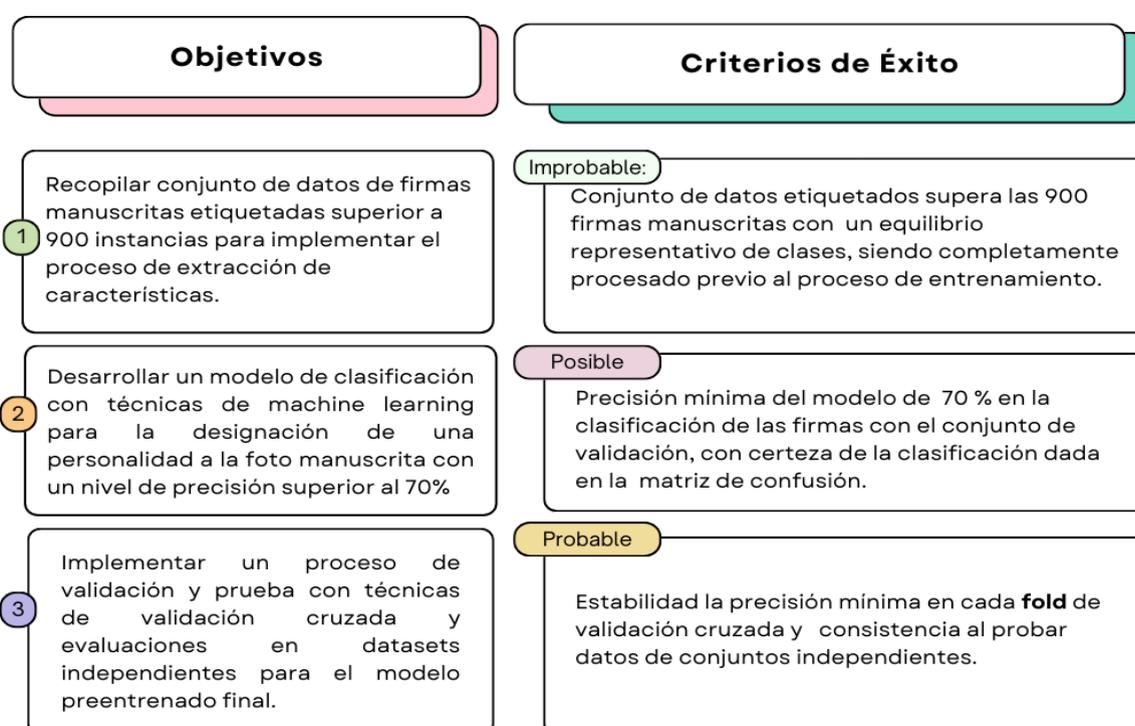


Figura 13 Objetivos y criterios de éxito de data mining

Para determinar la viabilidad de cada criterio, se consideró el análisis general del negocio, identificando el personal disponible y los recursos asignables al proyecto. Sin embargo, esto no modifica el alcance del proyecto.

La Figura 13 muestra los objetivos y criterios de éxito del proyecto. Los objetivos son recopilar más de 900 firmas manuscritas etiquetadas, desarrollar un modelo de *machine learning* con precisión superior al 70%, e implementar validación cruzada y pruebas en datasets independientes. Los criterios de éxito se clasifican como Improbable, superando 900 firmas con equilibrio de clases; Posible, logrando una precisión mínima del 70% en

validación; y Probable, asegurando estabilidad y consistencia en la precisión durante la validación cruzada y pruebas independientes.

g. Plan de proyecto

a. Alcance de plan de proyecto

En esta sección, a través de un documento en *Confluence*, se define el alcance del plan de proyecto basándose en las tareas previas. Una vez determinado el alcance, se desglosan las tareas con sus respectivas descripciones y la duración en semanas de cada fase. El alcance del proyecto se ilustra en la Figura 14, mientras que la Figura 15 presenta el plan detallado del proyecto.

Alcance

Debe tener:	<ul style="list-style-type: none">• Debe incluir todas las fases de la metodología CRIPS DM para el modelo de Grafología.• Debe contemplar los objetivos de la fase de <i>Entendimiento del negocio previa</i>
Podría tener:	<ul style="list-style-type: none">• Asignación a mas de un miembro del equipo la tarea• Priorización de las tareas
Fuera del alcance:	<ul style="list-style-type: none">• Documentación extensiva a propia de la metodología descrita• Fecha de cada tarea.• Cronograma

Figura 14 Alcance del proyecto

Como se muestra en el alcance, el plan sigue las fases descritas en la metodología, incluyendo tareas previas a este entregable. No se incluye un cronograma rígido debido a la dependencia de tareas como la recolección y etiquetado de datos, ya que una planificación excesiva podría resultar ineficiente y sobrecargar el flujo de trabajo con tareas bloqueadas. No se contempla agregar documentación adicional a la de este documento.

b. Descripción de tareas por fases

Las seis fases de la metodología CRISP-DM se muestran en la Figura 15.

Fase	Duración (Semanas)	Recursos	Tareas
Comprensión del negocio	2 SEMANAS	Grafólogo experto <hr/> Software para gestión de proyectos	<ol style="list-style-type: none"> 1. Reuniones con el grafólogo para definir alcance del proyecto 2. Documentación de requisitos, objetivos y recursos del proyecto
Comprensión de los datos	3 SEMANAS	Grafólogo experto <hr/> Dataset de firmas	<ol style="list-style-type: none"> 1. Conformación del dataset mayor a 900 firmas manuscritas. 2. Etiquetado de firmas en base a 4 clases de personalidad
Preparación de los datos	2 SEMANAS	Librerías de preprocesamiento de imágenes	<ol style="list-style-type: none"> 1. Limpieza de datos para corrección de registros erróneos 2. Adaptación multimedia al formato de entrada del modelo. 3. División de datos en conjuntos de entrenamiento, prueba y validación
Modelado	3 SEMANAS	Bibliotecas de machine learning: Tensorflow y Keras <hr/> GPU para entrenamiento y rendimiento del modelo	<ol style="list-style-type: none"> 1. Prueba y selección de modelos para <i>transfer learning</i> 2. Entrenamiento y validación de modelo. 3. Refinamiento de modelos y ajuste de hiperparámetros.
Evaluación	2 SEMANAS	Grafólogo experto	<ol style="list-style-type: none"> 1. Evaluación del modelo con las métricas F1-score, recall, precisión, pérdida y matriz de confusión 2. Revisión de resultados con etiquetado y predicción en tiempo real.
Despliegue	1 SEMANA	Servidor para despliegue	<ol style="list-style-type: none"> 1. Generación de una API documentada para consumir el modelo e integrarlo a sistemas externos

Figura 15 Tareas por fase de CRISP DM

Es importante destacar que se debe comprender un dataset mayor a 900 firmas para que se pueda asegurar una diversidad y representatividad de las cuatro clases de

personalidad. Además, tomando en cuenta el tiempo y personal disponible se considera este como el mayor número posible de firmas obtenibles.

Los recursos necesarios, la descripción de las tareas y su duración estimada en semanas también fueron contemplados en la descripción de las tareas. Aunque este plan de proyecto está documentado en Confluence, se ha creado un tablero Kanban para controlar el plan de trabajo. En la Figura 16 se encuentra el tablero con tareas en progreso, cada tarea se encuentra etiquetada con la fase de la metodología a la que pertenece. Ej. Definir alcance del proyecto: Comprensión del negocio.

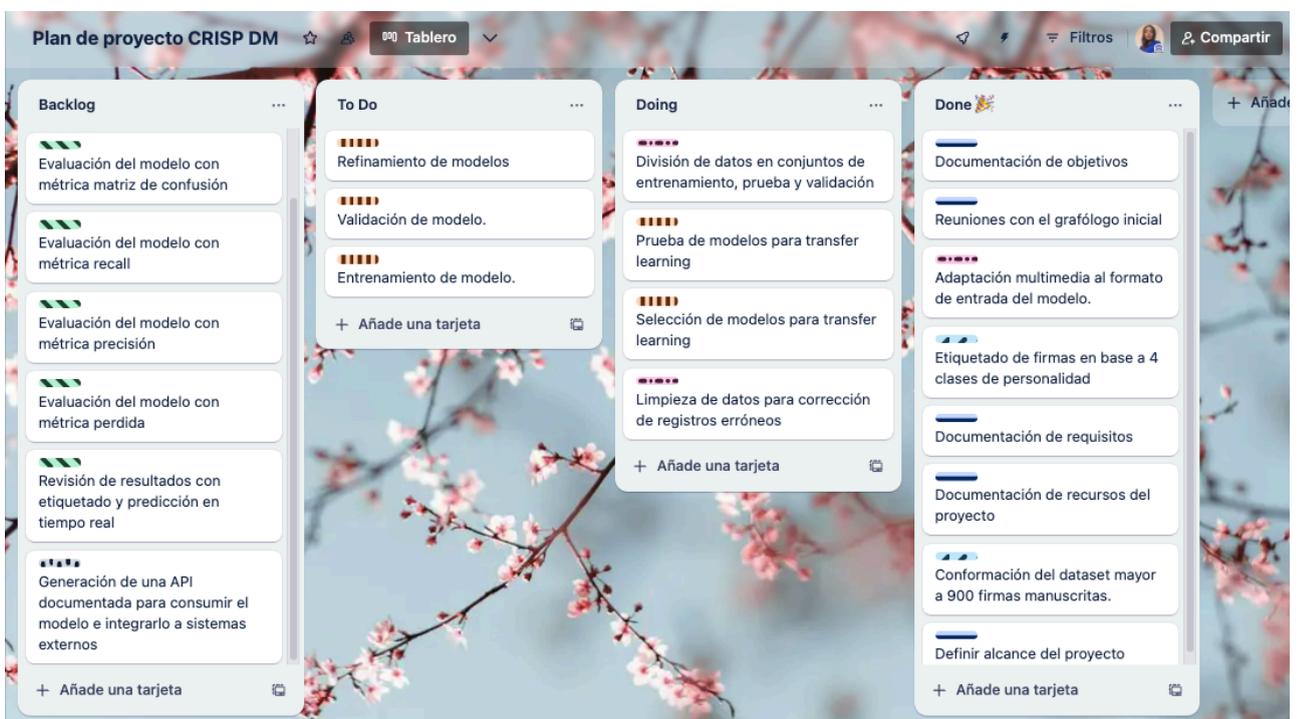


Figura 16 Tablero Kanban de gestión de plan proyecto CRISP DM

2.2.2 Fase 2: Comprensión de los datos

En esta sección esta contemplada la recopilación de la información requerida para resolver el problema y la exploración del tipo de datos y las características asociadas a los mismos como tamaño de muestra, calidad, dimensiones, etc.

Recolección de datos

Para el entendimiento de los datos se planteó inicialmente la posibilidad de utilizar *datasets* de firmas manuscritas disponibles en línea. En Repositorios públicos como

Kaggle se tiene una variedad de datasets que contiene 1021 firmas con 196.54 MB [37].



Figura 17 Ejemplo de Firma obtenida de repositorio Kaggle [36]

. Sin embargo, se presentaron las siguientes consideraciones que concluyeron en rechazar el uso del datasets.

- Las firmas representan la escritura de realidades sociales diferentes, ya que provienen de regiones orientales reconocibles.
- Algunos trazos de las firmas no eran consistentes entre sí, lo que diversificaba en exceso las muestras de un mismo dataset y podía causar errores en el modelo.

Como objetivo de esta tarea se construye una recopilado de imágenes manuscritas de la población universal que cumpla con las características descritas a continuación en Tabla 1.

Tabla 1 Características de datos recopilados

Consideraciones	Descripciones Recopilación	Cualidad a garantizar	Resultados
La mayoría de las imágenes del datasets representan formas de escritura no aplicables a la cultura donde se desarrolla nuestro público objetivo.	Matriz de 3x5 en dimensiones A4 con capacidad de 15 firmas.	<ul style="list-style-type: none"> • Orden • Consistencia 	Imágenes de firmas manuscritas etiquetadas en base a los cuatro temperamentos.
La orientación del <i>datasets</i> se enfoca en detección de información fraudulenta afectando el alcance planteado previamente.	Trazos realizados con bolígrafo de punta fina 0.7mm de tinta azul.	<ul style="list-style-type: none"> • Nitidez • Visibilidad • Contraste y diferenciación con líneas de matriz 	Los datos recopilados son representativos de la población universal mayor a 18 años, de cualquier género, sin discriminación de su ocupación y que garantice una amplia gama demográfica.
En ningún <i>datasets</i> se encontró etiquetas relacionadas al caso de estudio.	Escaneo de firmas con escáner estativo de firmas.	<ul style="list-style-type: none"> • Consistencia • Claridad • Manejo de archivos 	Las imágenes no están ligadas a información de personas, únicamente se toma la firma bajo conocimiento y autorización

Descripción de los datos

Al terminar el proceso de obtención de firmas se alcanzaron los siguientes entregables:

1. Biblioteca de 1134 imágenes que representan las muestras de firmas manuscritas.
2. Las imágenes guardadas en formato *PNG* con compresión mitigando pérdida de calidad
3. Imágenes con dimensiones de 2028 pixeles de ancho por 1782 pixeles de alto.
4. Imágenes nombradas con un identificador único bajo el esquema recolector-número (Ej. K001) después de estandarización serán guardadas como "F#"



La Figura 18 muestra el resultado de las firmas escaneadas. Se aseguró que los trazos sean de línea continua en color azul y que todas las imágenes tengan el mismo espacio en blanco alrededor de los trazos debido a la matriz impresa en la que fueron recolectadas. Esto ayudó al modelo a identificar las firmas por su tamaño y a asociarlas correctamente a las etiquetas de entrenamiento.

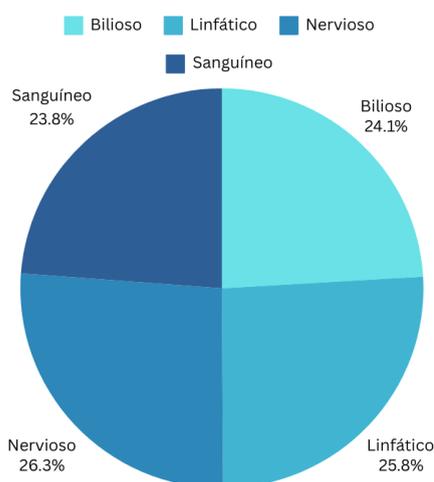
Exploración de los datos

Con la descripción previa de los datos, se llevó a cabo el análisis del etiquetado de datos. basado en el número de firmas asociadas a cada etiqueta en matriz de etiqueta generada por el grafólogo En la

Tabla 2 Firmas obtenidas por etiquetase presenta el número de firmas por cada etiqueta con una varianza de 2% entre los porcentajes de cada categoría. En la Figura 19 Diagrama de Distribución de firmas etiquetadas se evidencia la distribución existente por número de firmas sobre la muestra de 1134.

Tabla 2 Firmas obtenidas por etiqueta

Etiqueta	Número de Firmas
Bilioso	273
Linfático	293
Nervioso	298
Sanguíneo	270



Con los resultados previos se contempla las condiciones necesarias de los datos para solventar los casos especificados con sus respectivos objetivos. La Tabla 2 y la Figura 19 muestran la distribución de firmas obtenidas por etiqueta, clasificadas en cuatro categorías de personalidad: bilioso, linfático, nervioso y sanguíneo. La tabla indica que hay 273 firmas biliosas, 293 linfáticas, 298 nerviosas y 270 sanguíneas. El diagrama de distribución visualiza estos datos, mostrando que las firmas están bastante equilibradas entre las categorías, con un ligero predominio de las firmas nerviosas (26.3%) y linfáticas (25.8%), seguidas de las biliosas (24.1%) y sanguíneas (23.8%).

Verificación de la calidad de los datos

En esta etapa se encuentra el trabajo realizado en conjunto con la parte técnica y el grafólogo especialista. El personal técnico se encargó de verificar para cada categoría de imágenes recopiladas y etiquetadas la validación de calidad de las muestras, buscando que las imágenes obtenidas no presenten rasgos, sombras, manchas que puedan afectar el reconocimiento de trazos en el modelo.

Mientras que el grafólogo rectificó que cada imagen con calidad técnica necesaria sea admisible para análisis grafológico. Como segunda validación se realizó pruebas en vivo con una comparación entre etiqueta registrada en matriz de personalidad y el valor real expuesto por el grafólogo.

Con esto se trató de garantizar la mayor fiabilidad de los datos referentes a calidad, disminución de datos atípicos o errores y certeza de etiquetado correcto.

Preparación de los datos

En esta sección se acondicionó la información previa a nivel técnico para que cumplan con los requerimientos de entrada del modelo de *machine learning*. Se contempla preprocesamiento de mejora de imagen, aumento de frecuencia y adecuación de formato a una red neuronal convolucional.

2.2.3 Fase 3: Selección de datos

Aunque esta tarea indica plantear criterios de selección y exclusión de datos. Se tuvo en cuenta la doble validación de datos dada por técnico y grafólogo garantiza un porcentaje alto de viabilidad de los datos, por ello se utilizó el total de los datos validados que cumplen:

- Estándar de escaneo para todo el conjunto de datos
- Etiquetado de datos verificado por grafólogo
- Son legibles
- Son adecuados para el procesamiento
- Tiene variedad representativa de estilos, tamaños y formas
- Evaluación de distribución de etiquetado para garantizar equilibrio

Limpieza de Datos

Esta sección garantiza la confiabilidad y precisión del conjunto de datos, verificando la ausencia de errores, la consistencia de la información y la calidad general de los datos. Al tratarse de un conjunto de información generado propiamente por quienes conforman el proyecto y se ha aplicado procesos de validación continua en fases previas, no se presenta la necesidad de corregir ni tratar valores ausentes o incorrectos.

Estructura de Datos

Para optimizar el acceso y manejo de los datos a lo largo de todo el flujo de trabajo, se implementó una estructura organizada.:

1. Las imágenes de firmas manuscritas se nombraron inicialmente con la nomenclatura "F#" siguiendo su orden de ingreso en un único directorio general posterior a su validación.
2. La información de las etiquetas se consolidó en un archivo CSV, con estructura Código de Firma - Etiqueta.
3. Con Python, se automatizó la clasificación de las imágenes, distribuyéndolas en 4 directorios de acuerdo con sus etiquetas descritas en
4. Tabla 2 .

La

Tabla 2 presenta las etiquetas del conjunto de datos divididas en los conjuntos de entrenamiento, validación y prueba. Previamente, los datos pasan por un balanceador de clases que emplea una función lambda para evaluar la frecuencia de cada categoría.


```

filename1 = str(directory + filename)
img = cv2.imread(filename1)
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
img_denoised = cv2.fastNlMeansDenoising(gray, 30, 7, 21)
bilateral = cv2.bilateralFilter(img_denoised, 9, 150, 150)
adaptive_thresh = cv2.adaptiveThreshold(bilateral, 255, cv2.ADAPTIVE_THRESH_MEAN_C, cv2.THRESH_BINARY, 11, 4)
kernel = np.ones((4,4), np.uint8)
closing = cv2.morphologyEx(adaptive_thresh, cv2.MORPH_CLOSE, kernel, iterations=1)
kernel_sharpen = np.array([
    [-1, -1, -1],
    [-1, 10, -1],
    [-1, -1, -1]
])
sharpened_image = cv2.filter2D(closing, -1, kernel_sharpen)
resized = cv2.resize(sharpened_image, target_size, interpolation=cv2.INTER_AREA)

```

Figura 21 Preprocesamiento de firmas manuscritas

Además, el script de preprocesamiento se ha implementado un proceso de normalización y formateo de etiquetas. La normalización se aplica justo antes de introducir los datos al modelo para asegurar trato uniforme, independientemente de cualquier manipulación previa, y evitar la fuga de datos cuando se divide en conjuntos de entrenamiento, validación y prueba.

En la Figura 22 se describe el proceso de data *augmentation* para aumentar artificialmente el número de muestras mediante transformaciones definidas. El proceso de normalización se lleva a cabo tanto en ambos conjuntos de datos, dividiendo cada valor de píxel por 255 y escalándolo al rango [0,1], lo cual facilita la convergencia del modelo durante el entrenamiento.

```

train_datagen = ImageDataGenerator(rescale=1./255,
    rotation_range=rot_ran,
    width_shift_range=width_s_r,
    height_shift_range=height_s_r,
    horizontal_flip = hor_flip)
train_generator = train_datagen.flow(x_train, y_train_ohe, shuffle=False, batch_size=BATCH_SIZE, seed=seed)
# Crear generador para validación
val_datagen = ImageDataGenerator(rescale = 1./255)
val_generator = val_datagen.flow(x_val, y_val_ohe, shuffle=False, batch_size=BATCH_SIZE, seed=1)

```

Figura 22 Normalización, Validación y Data Augmentation

Se destaca que la generación de datos aumentados se aplica exclusivamente al conjunto de entrenamiento. Esto asegura que la evaluación del modelo se realice sobre imágenes nuevas y sin alteraciones.

```

#Convertir las etiquetas de clase de texto en etiquetas codificadas one-hot
y_train_ohe = pd.get_dummies(y_train.reset_index(drop=True)).values
y_val_ohe = pd.get_dummies(y_val.reset_index(drop=True)).values
y_test_ohe = pd.get_dummies(y_test.reset_index(drop=True)).values

```

La Figura 23 detalla el código empleado para la conversión a etiquetas *one-hot*. Por ejemplo, en el caso de las 4 clases existentes, una instancia de la clase 2 se representaría como [0, 0, 1, 0]. Finalmente, se transformaron las etiquetas categóricas de las clases a representaciones numéricas utilizando la técnica de codificación *one-hot*. De acuerdo con EITCA (2023), esto permite representar variables categóricas como vectores binarios, garantizando la compatibilidad entre los datos de entrada y el modelo, y simplificando el cálculo de las funciones de pérdida durante el entrenamiento.

2.2.4 Fase 4: Modelado

Selección de técnicas de modelado

Se eligió una técnica de aprendizaje profundo debido a su capacidad para extraer características de alto nivel, adecuada para la cantidad limitada de datos disponibles. Se usó una red neuronal convolucional (CNN) para la clasificación de imágenes, capturando su estructura espacial y patrones. Se aplicó transfer learning con InceptionV3, una CNN preentrenada en *ImageNet*, para aprovechar sus características visuales y realizar un ajuste fino en las capas superiores del modelo. InceptionV3 utiliza módulos estructurados para convolución, *pooling* y concatenación paralela, aumentando la capacidad de la red sin incrementar significativamente los parámetros.

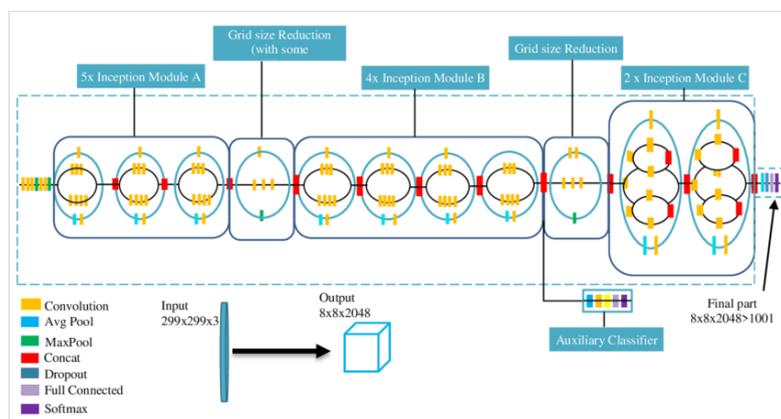


Figura 24 Arquitectura Inception V3 [37]

La **Error! Reference source not found.** muestra la arquitectura del modelo InceptionV3, destacando la combinación simultánea de convolución y pooling para reducir la dimensionalidad sin incrementar significativamente los parámetros. Incluye varios módulos Inception y un clasificador auxiliar para mejorar la precisión del modelo.

Diseño de pruebas

Se determino evaluar la red neuronal en términos de calidad y consistencia de *EarlyStopping*, que finaliza el entrenamiento cuando la métrica de validación no mejora, y *ReduceLROnPlateau* se evalúan los siguientes valores:

- **Accuracy:** Porcentaje de predicciones correctas.
- **Loss:** Desviación de la predicción respecto a los valores reales.
- **Confusion Matrix:** Precisión del modelo en cada clase, mostrando la frecuencia entre valores reales y predichos.
- **Recall:** Porcentaje de casos positivos identificados correctamente.
- **F1-Score:** Promedio armónico de accuracy y recall.

Para controlar el rendimiento, se utiliza técnicas como *EarlyStopping*, que finaliza el entrenamiento cuando la métrica de validación no mejora, y *ReduceLROnPlateau*, que reduce la tasa de aprendizaje automáticamente cuando las métricas de rendimiento disminuyen.

Construcción del Modelo

c. Estructura del modelo

Se definió un modelo basado en InceptionV3 preentrenado, añade capas densas, de normalización y de abandono, y compila el modelo para la clasificación de cuatro clases con el código de la Figura 25.

```

base_inception = InceptionV3(weights=pesos, include_top=False,
| | | | | | | | | | input_shape=(shapex, shapey, shapez))

base_inception.trainable = False
for layer in base_inception.layers:
|   layer.trainable = False
|   if layer.name == 'mixed7':
|       break

out = base_inception.output
out = Flatten()(out)
out = Dense(1024, activation='relu',
|   |   kernel_regularizer=regularizers.l2(0.00001))(out)
out = BatchNormalization()(out)
out = Dropout(0.5)(out)
total_classes = 4

predictions = Dense(total_classes, activation=activat2, name='predictions')(out)
#Creación y compilación del modelo
model = Model(inputs=base_inception.input, outputs=predictions)
opt1 = optimizers.SGD(learning_rate=learning_rate, momentum=moment, nesterov=True)
model.compile(loss=loss, optimizer=opt1, metrics=["accuracy"])

```

Figura 25 Estructura del modelo code

d. Early stopping

La Figura 26 muestra la configuración del criterio de parada temprana para interrumpir el entrenamiento si la pérdida de validación no mejora después de seis épocas, restaurando los mejores pesos.

```

early_stopping = EarlyStopping(
|   monitor='val_loss',
|   min_delta=0.0001,
|   patience=6,
|   verbose=1,
|   mode='min',
|   restore_best_weights=True
)

```

Figura 26 Early Stopping

e. Data Augmentation

En la Figura 27 se establece parámetros de aumento de datos, como rotación, desplazamiento y tamaño de lote, para crear variaciones de las imágenes de entrenamiento y aumentar la robustez del modelo.

```

batch_size = 48
rotation_range = 30
width_shift_range = 0.2
height_shift_range = 0.2
horizontal_flip = 'False'
seed = 57

```

Figura 27 Data Augmentation code

f. Parámetros para división de conjuntos

La Figura 28 especifica los tamaños y estados aleatorios de división de datos en conjuntos de prueba y validación, asegurando una proporción del 15% para cada uno.

```

#Parámetros para división de conjuntos
target_size_x = 299
target_size_y = 299
test_size = 0.15 #% para prueba
random_state = 42
test_size2 = 0.15 #% para validación
random_state2 = 42

```

Figura 28 Parámetros para división de conjuntos

g. Parámetros del modelo

La Figura 29 se configura los parámetros del modelo, incluyendo la forma de entrada, la función de activación, la tasa de aprendizaje y la ruta de guardado del modelo entrenado.

```

weights = 'imagenet'
input_shape_x = 299
input_shape_y = 299
input_shape_z = 3
activation = 'relu'
activation_pred = 'softmax'
loss = "categorical_crossentropy"
learning_rate = 0.0001
momentum = 0.8
batch_size = 48
epochs = args["epochs"]
model_path_save = 'models/transfer_inceptionV3.h5'
model_path_save_json = 'models/transfer_inceptionV3.json'

```

Figura 29 Parámetros del modelo

Evaluación del modelo

La evaluación preliminar se centra en cumplir los objetivos de la minería de datos, mientras que la siguiente sección abarcará los objetivos del negocio.

Bajo el plan de pruebas, se realizó el siguiente procedimiento:

1. Durante el entrenamiento, se aplicó *early stopping* para evitar el sobreajuste.
2. Se utilizó *reduce on plateau* para ajustar automáticamente la tasa de aprendizaje cuando la mejora del modelo se reduce.
3. El modelo final se evaluó con datos independientes, que no fueron utilizados en el entrenamiento ni en la validación, asegurando su capacidad de generalización.

Dada la naturaleza del problema, se busca mantener precisión y generalización en datos no vistos mediante el siguiente procedimiento:

a. Evaluación de los resultados

La fase de evaluación analiza críticamente todos los datos tratados durante el proyecto, midiendo la efectividad del modelo de clasificación de firmas manuscritas según la personalidad del usuario. El objetivo futuro es presentar a los usuarios un perfil de persona compatible psicológicamente para orientar su elección de parejas.

Los puntos clave son:

- Gráficas de precisión y pérdida en entrenamiento y validación.
- Métricas como *recall*, *F1-score*, pérdida y precisión.
- Matriz de confusión.
- Clasificación con imágenes de prueba.

Cada resultado será analizado en profundidad en el apartado 3 de resultados generales del proyecto.

b. Revisión del modelo

Este apartado revisa el proceso metodológico en su totalidad. Surgieron complicaciones debido a la diferencia entre los datos usados para entrenar InceptionV3 y el datasets. No obstante, las técnicas aplicadas permitieron alcanzar un nivel aceptable de precisión y pérdida para la cantidad de datos disponible.

Finalmente, el modelo cumple con los objetivos técnicos y del negocio, permitiendo clasificar la personalidad de una firma en 4 clases, con porcentajes de aproximación a cada una.

c. Próximos pasos

Para continuar el proyecto, se almacenan los formatos .h5 y .json del modelo para crear una API en Flask y facilitar su integración al sistema web. La aplicación se puso en contenedores de *Docker* y, con Render como Platform as a Service, se despliega la aplicación, garantizando disponibilidad total mediante soporte basado en contenedores.

2.2.5 Fase 5: Fase de Despliegue

Plan de despliegue

Se preparo el lanzamiento de la API para el modelo, trasladándolo de un entorno de desarrollo a uno de producción, utilizando herramientas e infraestructura en la nube. Las tareas clave son:

1. Preparación del Ambiente de Producción:

- Render como *Plataform as a Service*, con tecnologías de contenedores y configuración por defecto en rendimiento y seguridad.
- La API en *Flask* fue puesta en contenedores y almacenada en *GitLab* debido al tamaño de los archivos del modelo.

2. Gestión de dependencias:

- El contrato de servicio entre la aplicación se implementó con Docker, mediante Integración y despliegue continuos.

3. Documentación:

La API está documentada y almacenada en Gitlab, debido a su tamaño, para detallar las entradas y salidas del *endpoint* y su integración con Render, agilizando la operación y mantenimiento de la infraestructura en contenedores.

El servicio incluye el preprocesamiento de imágenes recibidas antes de ingresar al modelo, asegurando que todas las imágenes pasen por un proceso estandarizado.

Monitoreo

Tabla 3 Planificación del monitoreo del modelo

Aspecto	Descripción
---------	-------------

<p>Monitoreo del rendimiento</p>	<p>Con Render, se pueden acceder a métricas de rendimiento para monitorear el ciclo de consumo del modelo en producción, como el tiempo de respuesta de la API y posibles fallos en el servidor.</p>
<p>Mantenimiento y actualizaciones</p>	<p>Con proceso de CI/CD y la práctica <i>serverless</i>, las actualizaciones en el repositorio se despliegan automáticamente en el entorno de producción.</p>

2.2.6 Fase 6: Informe final

Para la generación del informe se toma los objetivos de la minería de datos y negocios enlazados con éxito al construir el modelo que predice la personalidad de un usuario con técnicas de ML, evaluación de sus métricas de rendimiento y despliegue para consumo público mediante la metodología CRISP -DM. Se considera que los resultados obtenidos ayudan al usuario a afianzar las decisiones en elección de pareja compatible psicológicamente en base a la actuación del subconsciente.

Figura 30 Resumen de informe final, se describen los eventos destacados para alcanzar los objetivos producidos en este marco metodológico:

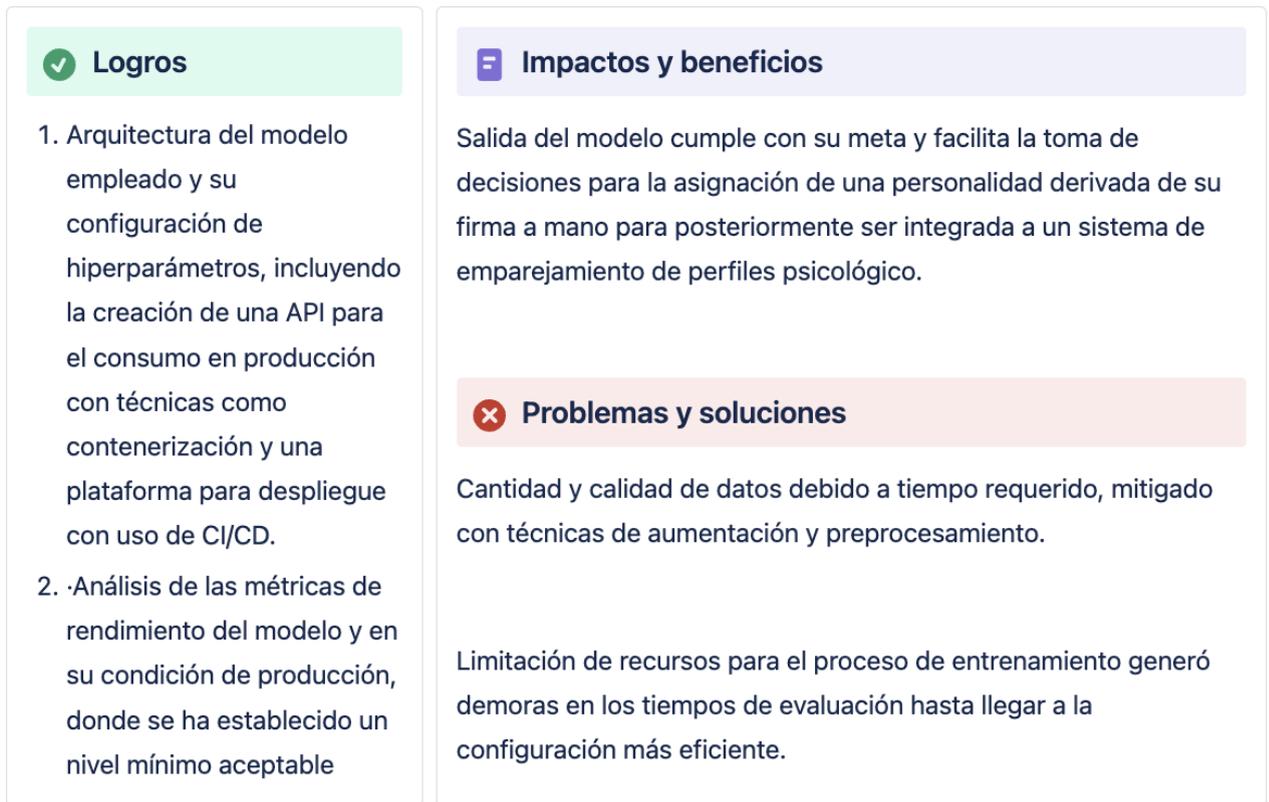


Figura 30 Resumen de informe final

Revisión del Proyecto:

En la revisión final del proyecto, se evalúan las acciones realizadas y sus resultados, para impulsar mejoras continuas en futuros proyectos. A pesar de la alta complejidad del problema de analizar firmas manuscritas para identificar patrones de personalidad, se utilizó *transfer learning* para mejorar el rendimiento del modelo. Aunque hubo desafíos como la obtención acelerada de datos y la calidad de las imágenes, se logró cumplir con los objetivos establecidos y se implementó una API para consumir el modelo de ML.

2.3 Metodología Kanban en proyectos de Software

Para el desarrollo de la segunda parte del proyecto se empleó la metodología ágil Kanban enfocada en proyectos de software basado en transparencia, eficacia y adaptación al cambio con un flujo de trabajo dinámico.

El proyecto se centra en desarrollar un sistema web que permita determinar la personalidad de un usuario a través de su firma manuscrita utilizando técnicas de grafología y un modelo de machine learning. Este sistema pretende ofrecer un servicio nuevo en Ecuador, donde la grafología no es muy utilizada. Además, el proyecto busca

proporcionar un análisis detallado de la personalidad y la compatibilidad entre dos personas, facilitando conexiones emocionales más profundas y confiables.

Los objetivos del proyecto incluyen implementar un sistema que permita a los usuarios cargar imágenes de sus firmas, proporcionar una interfaz intuitiva que permita visualizar métricas de compatibilidad y descripción de personalidad, evaluar el sistema mediante pruebas de aceptación del usuario ligadas al éxito del proyecto con casos de prueba, y desplegar el sistema web en internet para garantizar la disponibilidad del servicio. El alcance del proyecto abarca el diseño, desarrollo, pruebas y despliegue del sistema web, que incorporará todas las funcionalidades necesarias para su correcto funcionamiento.

a. Funcionalidades del sistema web:

- Registro y autenticación de usuario.
- Carga de imágenes de firmas manuscritas.
- Procesamiento de imágenes con modelo de clasificación.
- Análisis y visualización de resultados de personalidad.
- Comparación de personalidades entre dos usuarios.
- Mantener la privacidad y seguridad de los datos de usuarios.

b. Tecnologías empleadas:

Tabla 4 Tecnologías utilizadas para el desarrollo

Arquitectura	Microservicios
<i>Backend</i>	Nest.js
<i>Frontend</i>	React.js y Material UI
Despliegue <i>Frontend</i>	Netlify
Despliegue <i>Backend</i>	Entorno controlado
Gestión de Proyecto	Jira y Confluence

La Tabla 4 detalla las tecnologías utilizadas para el desarrollo del proyecto. En la arquitectura de backend se emplea Nest.js, mientras que para el frontend se utilizan React.js y Material UI. El despliegue del frontend se realiza en Netlify y el backend en un

entorno controlado. Para la gestión del proyecto, se utilizan Jira y Confluence, lo que facilita la organización y seguimiento del desarrollo.

2.3.1 Cálculo del Work in Progress (WIP)

Para calcular el WIP en el contexto de Kanban, se debe tener en cuenta la capacidad del equipo y el número de tareas en curso. Al momento de desarrollo del proyecto se contó con tres personas técnicas disponibles para cada tarea.

Se realizó un análisis que determinó que cada persona puede trabajar en dos tareas simultáneamente. Dado que disponemos de tres personas en el equipo, el número máximo de tareas en progreso (WIP) para el proyecto es seis. Este enfoque garantiza que el equipo no se sobrecargue y que las tareas se completen de la manera más eficiente posible.

Adaptación del tablero Kanban

Los tableros Kanban estándar tienen las columnas definidas de “Por hacer”, “En proceso” y “Terminado”. Sin embargo, el flujo de trabajo en software es más extenso y complejo por lo que se adaptó las columnas y su definición bajo el siguiente esquema:

Fases	Columnas Kanban	Descripción
TO DO 	Por hacer	Almacén de todas las tareas establecidas en el proyecto, pero aún no iniciadas. Incluye historias de usuario, requisitos, correcciones, etc.
DOING 	Diseño	Las tareas se encuentran en fase de análisis y diseño donde se planifica cómo se llevará a cabo la solución previa a su implementación, con la generación de esquemas y diagramas que respondan a las especificaciones técnicas y funcionales establecidas.
	Desarrollo	Fase que implementa las historias de usuario y requerimientos mediante la codificación de los diseños aprobados.
	Bloqueo	Comprende las tareas que se encuentren temporalmente detenidas por razones técnicas, legales o dependencias externas, sin embargo, seguirán activas.
	Prueba	Al finalizar el desarrollo se realizan pruebas de aceptación de usuario para asegurar el cumplimiento de requisitos en el flujo de casos de prueba
	Despliegue	Una vez concretadas y revisadas las funcionalidades del sistema en entorno de desarrollo serán migradas a producción para un consumo público.
DONE 	Terminado	Las tareas que han completado su fase de lanzamiento a producción tras ser revisadas están listas para ser entregadas al cliente.

Figura 31 Descripción columnas Kanban

La Figura 31 describe las columnas Kanban utilizadas en el proyecto. La columna "Por hacer" almacena todas las tareas establecidas, pero no iniciadas, incluyendo historias de usuario y requisitos. La columna "Diseño" contiene tareas en fase de análisis y diseño, donde se planifican soluciones y se generan esquemas y diagramas. La columna "Desarrollo" implementa las historias de usuario y requerimientos mediante la codificación. "Bloqueo" comprende tareas detenidas temporalmente por razones técnicas o dependencias externas. "Prueba" se dedica a las pruebas de aceptación de usuario tras la codificación del sistema. "Despliegue" incluye la migración de funcionalidades desde el entorno de desarrollo a producción. Finalmente, "Terminado" incluye tareas que han completado su fase de lanzamiento y están listas para ser entregadas al cliente.

Métricas de rendimiento

Los indicadores de rendimiento que se establecieron para medir la eficiencia del flujo de trabajo definido para el proyecto en base a los objetivos están:

1. Tiempo de ciclo: Tiempo promedio en el que tardan las tareas en pasar de estado “To Do” a “Done” pasando por pruebas exitosas y despliegue.
2. Tasa de flujo: Número de tareas completadas dentro del periodo observable para verificar la velocidad.

Historias de usuario HU

En la fase de análisis del negocio se utilizó como base para determinar los requisitos del proyecto que fueron plasmados como historias de usuario con criterios de aceptación que guiaran el desarrollo según tareas especificadas en el tablero Kanban en Jira, las historias creadas poseen un alto nivel de granularidad suficiente para guiar el desarrollo sin tener requerimientos funcionales más específicos.

Cada historia maneja el formato de código, título, prioridad, descripción y criterios de aceptación. Para el desarrollo del sistema web que cumpla todas las funcionalidades se tienen 6 HU almacenadas en *Confluence*, a continuación, La historia de usuario 6 está directamente relacionada con la funcionalidad de compatibilidad psicológica y se presenta con las HU globales del sistema en el Anexo II.

Tareas

Mediante la herramienta Jira se incluyen todas las tareas diseñadas para cumplir con el desarrollo del sistema web que cumpla con las historias de usuario. En el *backlog* se incluyen todo el conjunto de tareas con asignación a la persona responsable y por prioridad definida. Apoyado por el *wip* calculado y *Confluence* como almacenamiento de conocimiento el proyecto queda determinadas las siguientes tareas:

Tabla 5 Tareas ligadas a las historias de usuario

Historia de usuario	ID Tarea	Tarea
HU-01	TR-01	Diseñar la interfaz de usuario para el registro mediante <i>Google Accounts</i> o correo electrónico.
	TR-02	Conectar el formulario de registro hacia <i>Firebase Authentication</i> para gestionar la autenticación de usuarios.
	TR-03	Validar los formularios de entrada en el cliente para verificar formato de correo, requisitos mínimos de contraseña y campos nulos.
HU-02	TR-04	Diseñar la interfaz de carga de información con foto de firma manuscrita para completar el perfil de usuario.

	TR-05	Implementar la lógica del procesamiento de imágenes y datos personales para creación de perfil en la base de datos.
HU-03	TR-06	Desarrollar la interfaz de un <i>dashboard</i> para mostrar los resultados de la personalidad de la firma en la página principal.
	TR-07	Integrar y consumir el modelo de clasificación de firmas en un microservicio para enviar los porcentajes de compatibilidad al cliente.
	TR-08	Mostrar información detallada de la personalidad como tipo de temperamento, descripción, sexualidad, competencias, valores asociados al temperamento e información complementaria.
HU-04	TR-09	Diseñar la sección para la visualización de la rama académica y su descripción asociada a su personalidad.
	TR-10	Diseñar la sección para la visualización de un listado de las competencias vinculadas a la personalidad
HU-05	TR-11	Diseñar las tarjetas de recomendación vocacional para mostrar una descripción y sus competencias asociadas
	TR-12	Desarrollar un microservicio de lógica vocacional donde se genere el algoritmo de <i>matching</i> entre competencias del usuario y competencias de las carreras.

Flujo de trabajo

Jira permite modificar el flujo de trabajo que cada tarea debe seguir a lo largo de cada columna del tablero Kanban mediante la determinación de un flujo trabajo personalizado a las necesidades del proyecto. Es así como se construyó un flujo de trabajo que permita el intercambio de tareas entre cada fase, impidiendo el cambio de tareas que salten fases de validación como testeo.

Bajo esto queda de resultado el siguiente flujo de trabajo:

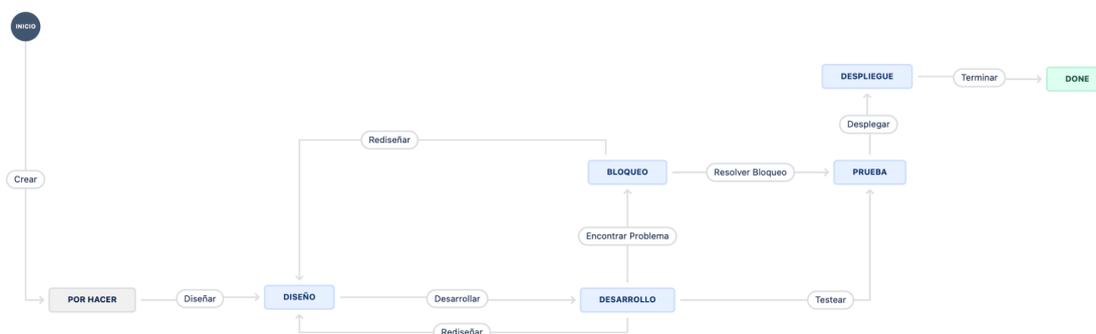


Figura 32 Flujo de trabajo definido en Jira para tablero Kanban

La Figura 32 se encuentra el flujo de trabajo definido en Jira para un tablero Kanban. Comienza con la creación de tareas en la columna "Por hacer", sigue con el diseño y desarrollo de las tareas. Si hay problemas, las tareas pueden pasar a la columna "Bloqueo" y volver a la fase de "Diseño" o "Desarrollo". Una vez resueltos, las tareas se prueban y, si son exitosas, se despliegan. Finalmente, las tareas completadas se marcan como "Terminado". Este flujo asegura una gestión organizada de las tareas desde su inicio hasta su finalización y la adaptación al cambio que puede presentar el proyecto.

En la Figura 33 se ilustra las columnas existentes en Jira y el flujo que siguen las tareas a lo largo del tablero. Cada tarjeta esta identificada por el responsable a cargo y un color que identifica la historia de usuario bajo lo que se desarrolla la tarea.

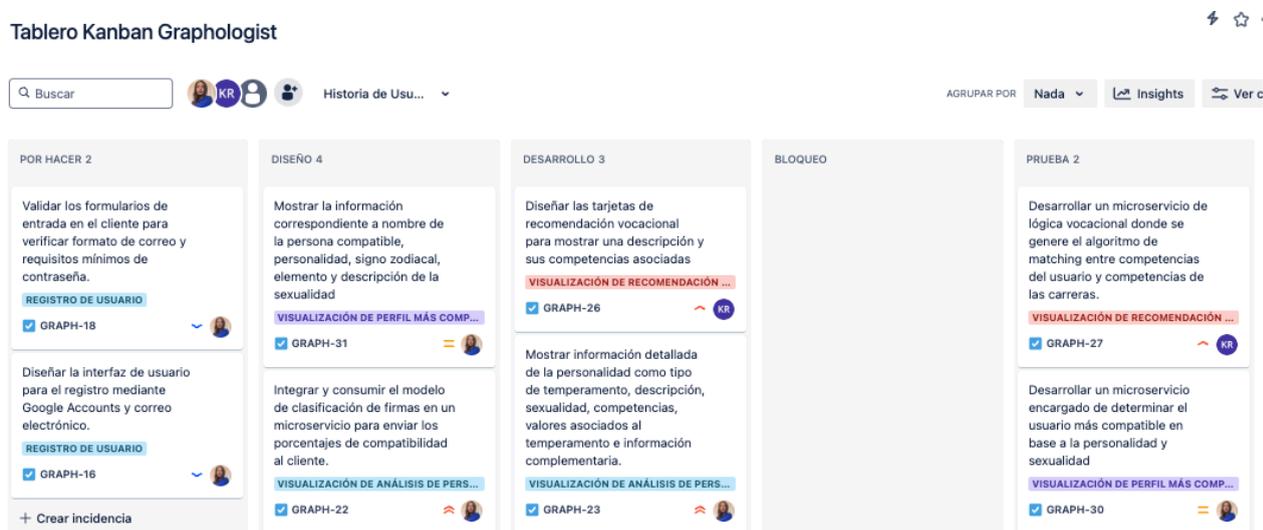


Figura 33 Tablero Kanban en Jira

Análisis y diseño

Este es uno de los componentes críticos en el desarrollo del sistema pues enmarca el análisis y diseño de los diferentes aspectos como arquitectura, diseño de bases de datos, diagramas de clases que puedan guiar el desarrollo en las siguientes fases, con el objetivo de promover un desarrollo ágil minimizando los riesgos

Diagrama de base de datos

En base al diagrama de clases y con el fin de soportar operaciones de almacenamiento y recuperación de información relacionada con los usuarios, firmas y análisis de personalidad, se aplicó el gestor de bases de datos MySQL. Este gestor generó el

esquema

mostrado

en

la

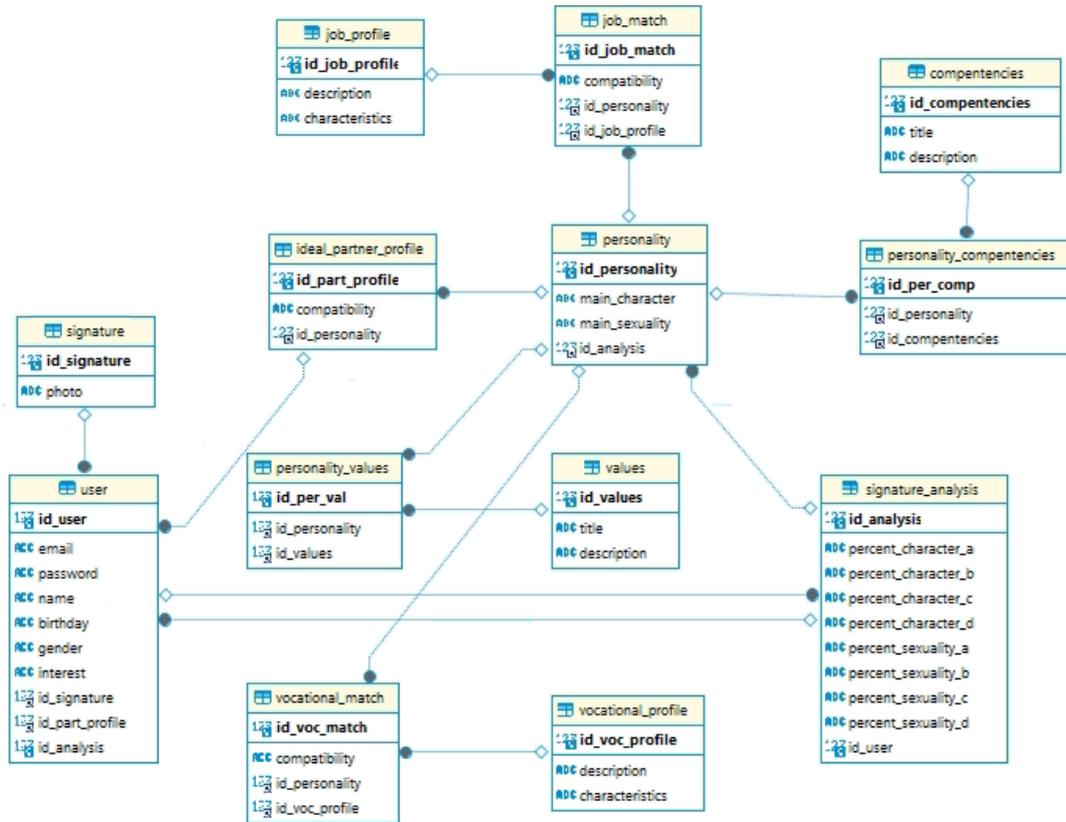


Figura 34, el cual satisface la necesidad de persistencia del sistema mediante la implementación con TypeORM.

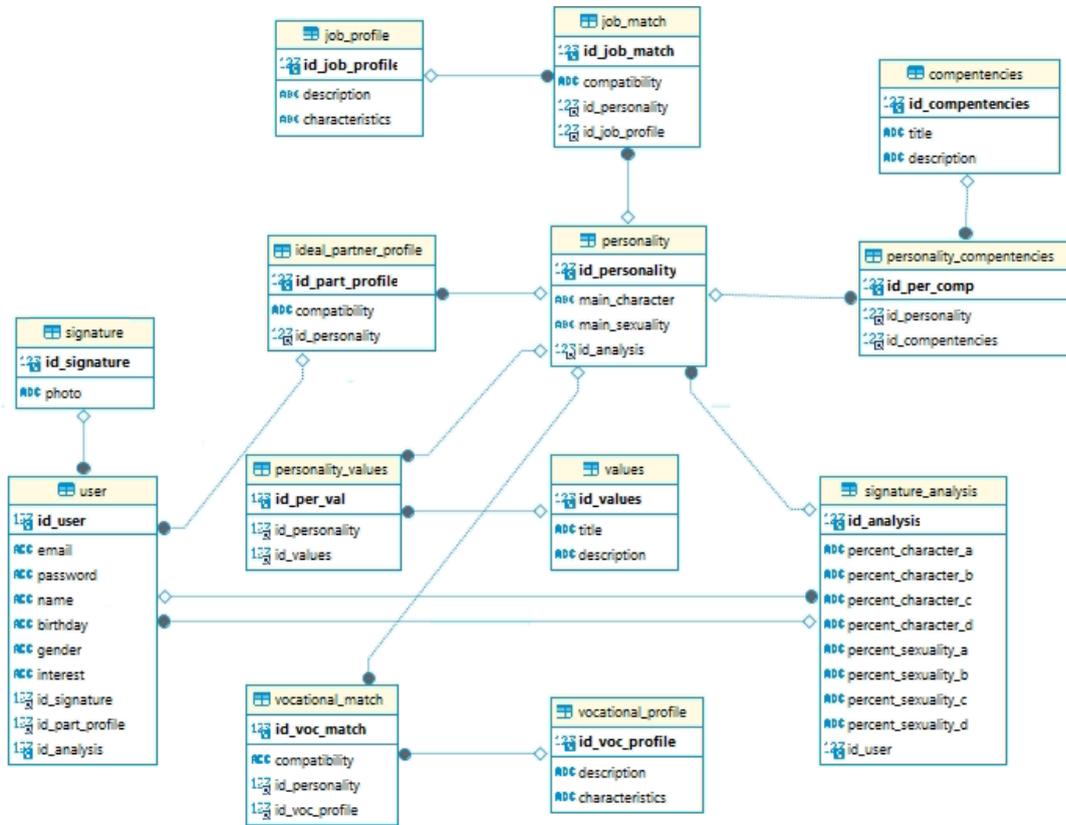


Figura 34 Modelo Entidad-Relación a partir de TypeORM

Arquitectura de microservicios

La estructura del sistema se basa en una arquitectura de microservicios utilizando Nest.js, lo que garantiza un desarrollo modular e independiente para facilitar el crecimiento, escalabilidad y mantenimiento de la aplicación. Se definieron cuatro microservicios y un *backend* general para separar la lógica de negocio, resultando en un sistema más organizado y eficiente. La Figura 35 muestra la arquitectura global, detallando la segmentación del *backend* con microservicios, *frontend* y el despliegue del sistema.

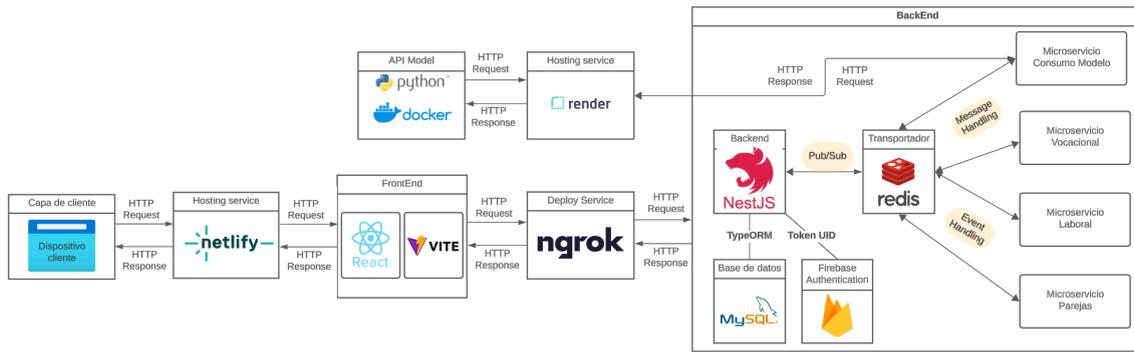


Figura 35 Diagrama de arquitectura del sistema web

Prototipos de interfaz de usuario

Mediante *Figma* se diseña un prototipo del sistema web que incluya diseño de experiencia de usuario para su implementación con retroalimentación temprana que permite realizar reajustes. En el Anexo II se encuentra el vínculo de vista del prototipo de uso del usuario.

Ejecución

a. Arquitectura de microservicios

Usando Nest.js que es un *framework* usado para la separación de componentes funcionales independientes que simplifica el proceso de desarrollo y mantenimiento aislado. Por diseño se determinan los siguientes microservicios:

- Consumo de modelo de machine *learning*.
- Módulo de recomendación vocacional.
- Módulo de compatibilidad de parejas.
- Módulo de recomendación de cargo laboral.

Se utilizó Redis para establecer la comunicación entre los microservicios y almacenar consultas en caché, agilizando así el envío de información entre servicios. Además, se desarrolló un *backend* global que gestiona operaciones generales complementarias.

b. Integración modelo de grafología

Se integró al sistema un modelo de ML para el procesamiento de imágenes y un algoritmo de redes neuronales convolucionales a través de una API creada con *Flask* y desplegada en Internet. Este microservicio se encarga de proporcionar los resultados de probabilidad de cada etiqueta a los módulos que lo necesiten.

c. Desarrollo Front End

Para desarrollar las vistas de la interfaz de usuario, se utilizó React.js junto con Material UI que es una biblioteca de componentes específica para *React*. Se siguió la estructura definida en el prototipo, incluyendo todas las interacciones necesarias entre las vistas. Aunque el diseño final puede diferir del prototipo debido a limitaciones de tiempo, se mantuvieron las funcionalidades e interacciones clave. Para manejar la información enviada desde el *backend*, se implementó *Redux*, una biblioteca de gestión del estado predecible para JavaScript, lo que permitió tener mayor control sobre los estados independientes de cada módulo.

d. Control de versiones y despliegue

Para todo el proceso de desarrollo se utilizó *Gitlab* para almacenar el código de los diferentes componentes del proyecto:

- API del Modelo: *Flask*
- *Backend*: Nest.js
- *Frontend*: React.js

Se crearon distintas ramas para cada repositorio, y se unieron mediante *commits* descriptivos para un control de versiones más eficaz. *Gitlab* está integrado con Jira, lo que permite relacionar las tareas definidas con el desarrollo y gestionar el avance del equipo de manera eficiente.

Gitlab permite almacenar una cantidad considerada de archivos a un costo reducido, por lo que se utiliza para almacenar las firmas etiquetadas y el modelo de predicción, que son archivos de gran tamaño.

Testing y Despliegue

Para esta fase, se realizaron pruebas de aceptación de usuario para abarcar los objetivos y expectativas del usuario final, seleccionando tres usuarios cercanos al perfil del grafólogo profesional que etiquetó el *dataset*. Se desarrollaron dos escenarios de prueba, documentados detalladamente, y se realizaron pruebas Beta en un entorno de producción controlado, utilizando Google Forms para recoger y analizar la retroalimentación.

Para el despliegue, se utilizó Ngrok para facilitar el acceso remoto al *backend* desarrollado en Nest.js como muestra la Figura 36, proporcionando conexiones HTTPS

cifradas. *Netlify* se empleó para el despliegue del *frontend*, ofreciendo una infraestructura robusta con integración continua como muestra la Figura 37, certificados SSL automáticos y alta disponibilidad. La configuración de ambos servicios permitió emular un entorno de producción, facilitando las pruebas de funcionalidad y demostraciones efectivas.

```
Tunnel Status      online
Version           2.1.3
Region            United States (us)
Web Interface      http://127.0.0.1:4040
Forwarding         http://05ae780c.ngrok.io -> localhost:4000
Forwarding         https://05ae780c.ngrok.io -> localhost:4000

Connections
  ttl    opn    rt1    rt5    p50    p90
  398    2      0.00  0.00  60.04  162.55
```

Figura 36 Configuración Ngrok

```
Deploy log [Preview] [Refresh] [Up] [Down] [Maximize log]
  > Initializing [Complete]
  > Building [Complete]
  > Deploying [Complete]
  > Cleanup [Complete]
  v Post-processing [Complete]

11047 7:41:02 PM: Starting post processing
11048 7:41:02 PM: Skipping form detection
11049 7:41:02 PM: Post processing - header rules
11050 7:41:02 PM: Post processing - redirect rules
11051 7:41:02 PM: Post processing done
11052 7:41:02 PM: Section completed: postprocessing
11053 7:41:03 PM: Site is live 🎉
```

Figura 37 Log de Deploy Netlify

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Resultados

Los resultados alcanzados durante el proyecto aplicando las metodologías CRISP-DM y Kanban con sus respectivos entregables se detallan en esta sección.

Metodología CRISP-DM

a. Precisión vs Perdida

Los resultados obtenidos al valorar el comportamiento de la precisión y pérdida en el entrenamiento y validación según las épocas avanzadas se presentan en la Figura 38.

Métrica	Comportamiento en Entrenamiento	Comportamiento en Validación
Precisión	Aumenta gradualmente, alcanza máximo de 73%.	Fluctuaciones notables, mínimo de 65%.
Pérdida	Disminuye de manera constante.	Mayor variabilidad, tiende a estabilizarse hacia el final en un valor menor.
Análisis	La mejora en la precisión de entrenamiento indica aprendizaje continuo, pero las fluctuaciones en la validación sugieren posible sobreajuste.	La reducción constante de la pérdida de entrenamiento sugiere optimización adecuada del modelo, aunque la validación podría indicar problemas de generalización. Utilización de <code>ReduceLROnPlateau</code> para ajustar la tasa de aprendizaje.

Figura 38 Resultados de pérdida y precisión

En la Figura 39 se muestra la evolución del rendimiento de precisión y pérdida durante el entrenamiento y la validación utilizando la arquitectura de InceptionV3 con ajuste fino y modificaciones de hiperparámetros.

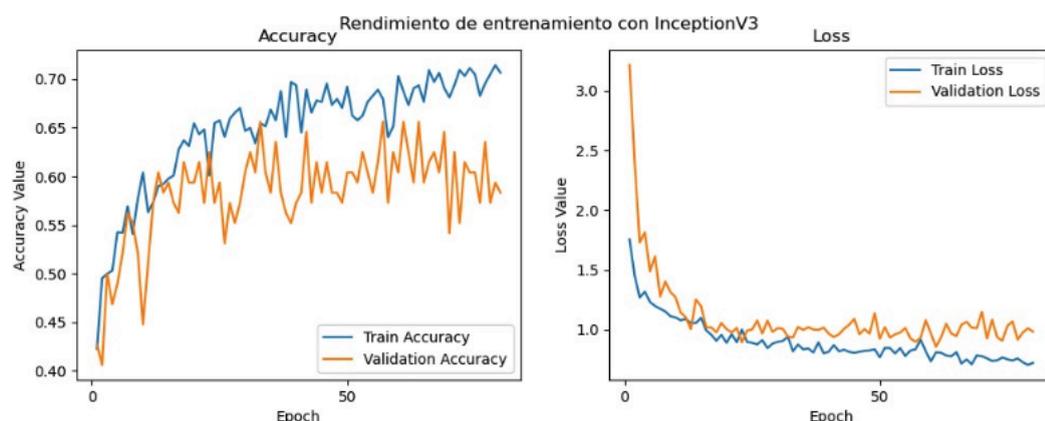


Figura 39 Rendimiento precisión y pérdida en entrenamiento y validación

De la Figura 39 se observa que la precisión del modelo en el conjunto de entrenamiento mejora consistentemente, alcanzando aproximadamente un 70%, lo que indica un buen aprendizaje. Sin embargo, la precisión en el conjunto de validación es más variable y se estabiliza alrededor del 60%, sugiriendo algún grado de sobreajuste. La pérdida en el conjunto de entrenamiento disminuye de manera constante, indicando una buena optimización, mientras que en el conjunto de validación también disminuye, pero con fluctuaciones. Esto puede deberse a diferencias entre los datos de entrenamiento y validación, o a la necesidad de ajustar hiperparámetros para mejorar la generalización del modelo. En resumen, el modelo aprende bien en el entrenamiento, pero necesita mejorar en su generalización y estabilidad en el conjunto de validación.

b. Recall y F1-Score

Para evaluar estas métricas, se emplean los resultados de la Figura 40 que detallan los valores individuales por cada clase y un promedio ponderado en cada columna.

	precision	recall	f1-score	support
Bilioso	0.72	0.71	0.71	62
Linfatico	0.68	0.75	0.71	63
Nervioso	0.74	0.72	0.73	63
Sanguineo	0.78	0.75	0.76	62
accuracy			0.72	248
macro avg	0.73	0.71	0.71	248
weighted avg	0.73	0.71	0.72	248

Figura 40 Recall y F1- score

A partir de la Figura 40 se obtienen los resultados del modelo de clasificación de firmas que muestran que el modelo tiene una precisión general del 72% (*accuracy*). La precisión, *recall* y f1-score de las cuatro categorías de temperamentos varían ligeramente: el temperamento sanguíneo tiene la mayor precisión (0.78) y f1-score (0.76), indicando que este temperamento es el más fácilmente reconocible por el modelo. El temperamento linfático tiene la menor precisión (0.68), lo que sugiere que el modelo tiene más dificultades para clasificar correctamente las firmas de esta categoría. El promedio ponderado de precisión, *recall* y f1-score es consistente en 0.73, lo que indica que el modelo tiene un desempeño equilibrado en general, aunque hay margen de mejora, especialmente en la clasificación de las categorías menos precisas.

A partir de esta tabla se determinan los siguientes resultados:

Métrica	Comportamiento en Entrenamiento	Comportamiento en Validación
Recall	Valores específicos por clase y ponderado final.	Indicativo de conservadurismo del modelo si alta precisión y bajo recall.
F1-Score	Media armónica entre precisión y recall.	Punto de equilibrio entre precisión y recall para identificar desviaciones.

Figura 41 Resultados de Recall Y F1-Score

c. Matriz de Confusión

bilioso	19	4	3	1
mfático	3	16	1	10
rvioso	4	1	22	3
guineo	0	2	1	24
	bilioso	mfático	rvioso	guineo

Figura 42 Matriz de confusión obtenida

En la Figura 42 se presenta la matriz de confusión obtenida del modelo donde las clases predichas están en el eje horizontal y las clases reales en el eje vertical. A partir de esto se obtiene el siguiente análisis:

Clase	Predicciones Correctas	Desviaciones en Clases Erróneas	Observaciones
Bilioso	19	8	Confusión moderada, especialmente con Linfático.
Linfático	16	10	Confusión notable con Sanguíneo en 10 casos.
Nervioso	22	2	Mejor tasa de clasificación correcta, solo 2 desviaciones.
Sanguíneo	24	2	Buena clasificación con solo 2 clases erróneas.

Figura 43 Resultados de matriz de confusión

La Figura 43 muestra el desempeño del modelo de clasificación de firmas para cada una de las clases de temperamentos: bilioso, linfático, nervioso y sanguíneo. Para la clase bilioso, hubo 19 predicciones correctas y 8 desviaciones, con una confusión moderada, especialmente con la clase linfático. La clase linfático tuvo 16 predicciones correctas y 10 desviaciones, presentando una confusión notable con la clase sanguíneo en 10 casos. La clase nervioso obtuvo la mejor tasa de clasificación correcta con 22 predicciones correctas y solo 2 desviaciones. Finalmente, la clase sanguíneo mostró una buena clasificación con 24 predicciones correctas y solo 2 desviaciones.

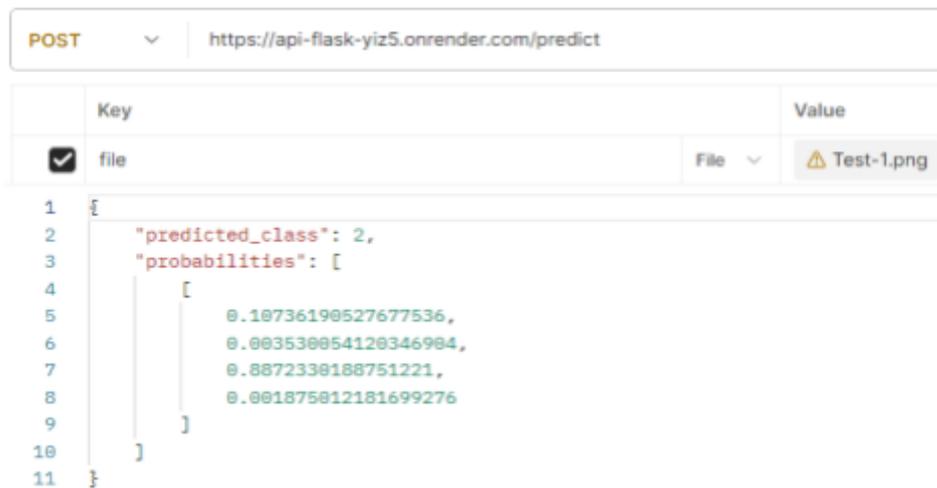
Se determinó que las etiquetas de nervioso y sanguíneo tienen mayores clasificaciones correctas, lo que no ocurre con las etiquetadas de bilioso y linfático lo que puede indicar un fallo al reconocimiento de características específicas de estas personalidades por el modelo.

El fallo muestra que el modelo tiene dificultades para clasificar correctamente las personalidades bilioso y linfático. Mientras que nervioso y sanguíneo tienen mayores tasas de acierto, los errores frecuentes en bilioso y linfático sugieren deficiencias en el modelo. Esto indica la necesidad de mejorar el entrenamiento del modelo mediante la recolección de más datos o el ajuste de los parámetros del modelo.

API Publicada

La aplicación *Flask API* se implementó exitosamente en Render, con todo su entorno contenerizado. Se configuró un único *endpoint* denominado *"/predict"* utilizando el método POST. La Figura 45 ilustra el consumo del servicio alojado mediante *Postman*. Esta API

recibe una imagen de firma en cualquier formato dentro del cuerpo de la solicitud, utilizando la clave "file", y devuelve un archivo JSON que incluye la clase de mayor predicción y las probabilidades asociadas a cada etiqueta.



```
POST https://api-flask-yiz5.onrender.com/predict

Key: file (checked) | Value: Test-1.png (File)

1 {
2   "predicted_class": 2,
3   "probabilities": [
4     [
5       0.10736190527677536,
6       0.003530054120346904,
7       0.8872330188751221,
8       0.001875012181699276
9     ]
10  ]
11 }
```

Mediante Postman Metodología Kanban

FronD-End conectada a Back end

La aplicación web se encuentra desplegada en Internet en <https://graphologist.netlify.app/> donde se presenta la interfaz de la Figura 46 donde se puede ver la información de dos usuarios como nombre, personalidad y descripción con su respectivo avatar y un grado de compatibilidad entre ellos.

a. Pantalla de Login que incluye Google Sign In



Figura 46 Login Graphologist

b. Pantalla de formulario interactivo para subir firma e información personal

Logo: *marcelo firma*

Bienvenido, Marcela

VOCAACIONAL

PARERIA IDEAL

CERRAR SESIÓN

Antes de comenzar

Responde las siguientes preguntas:

Género

Elige una opción:

Masculino

Femenino

Firma

Sube la foto de tu firma:

test.png

SUBIR IMAGEN

Edad

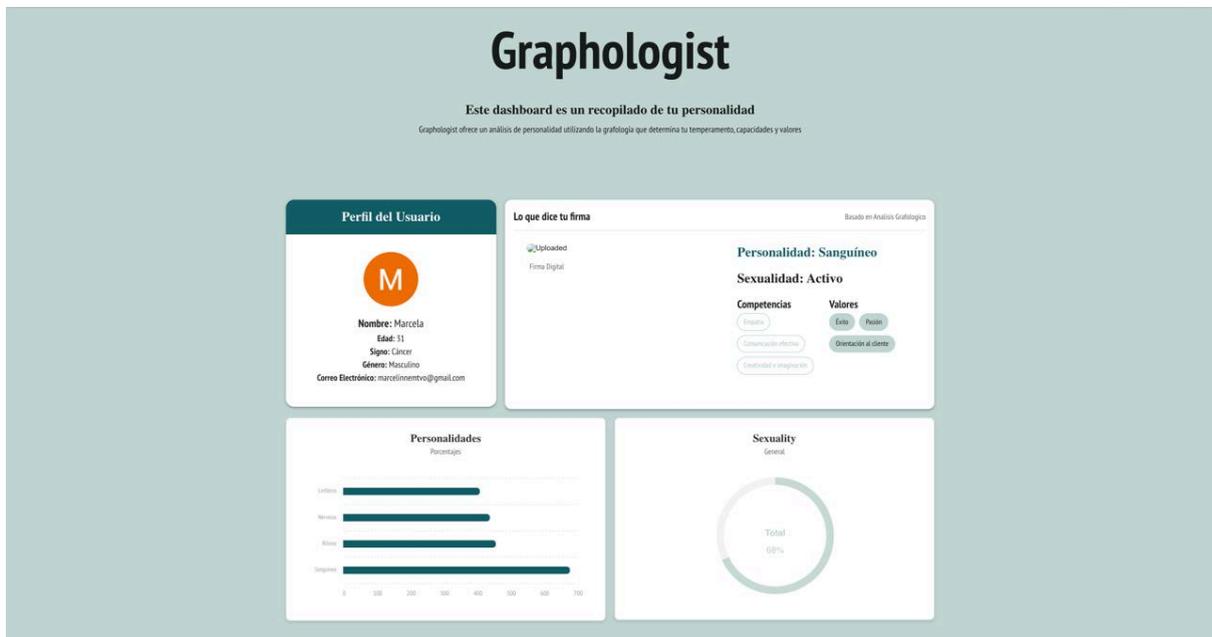
Elige tu fecha de nacimiento:

Fecha de Nacimiento: 07/19/1999

GUARDAR

Figura 47 Formulario de ingreso de datos

c. Pantalla de Dashboard con resumen de personalidad



Pantalla de compatibilidad entre usuarios



Figura 49 Persona compatible psicológicamente

Autenticación con Firebase

Mediante *Firebase* se utiliza su servicio de autenticación por lo que los usuarios pueden crear una cuenta e iniciar sesión mediante Google. En la Figura 50, se evidencia la gestión de usuarios realizado desde *Firebase* con los sujetos de prueba.

Identificador	Proveedores	Fecha de creación ↓	Fecha de acceso	UID de usuario
alvaradosteven315@g...		6 jul 2024	13 jul 2024	Lvh85y8gte00TFoMsCwga0jx...
kevinrivadeneira645@g...		6 jul 2024	6 jul 2024	g5hSYJJIKpNTu7r0XJhSpuCw...
sofia.soria.chamba@g...		1 jul 2024	1 jul 2024	wau5swDo9Ggl7lQqsUrpN2...
samu@gmail.com		30 jun 2024	30 jun 2024	5dfeXUCVujSSZV5fDfy9T8l2T...
salvarado@sodigsa.com		27 jun 2024	27 jun 2024	mELHtRcaFJdCyU0U9W7PFea...
daniel.velin.2001@gma...		25 jun 2024	25 jun 2024	2VTP9YpJl6f5ZMkctW0V8hp...
pedricles2000@gmail.c...		24 jun 2024	24 jun 2024	IA2qWnj7riSFmD1avSH01wW...
evita174123@gmail.com		24 jun 2024	24 jun 2024	Xkx2lj4mg0P8rGsWe40ulDnZ...
facturasogia0@gmail.c...		24 jun 2024	24 jun 2024	oCLQrnl9fjRPY9WpLn4uySVfG...
marcela@gmail.com		24 jun 2024	24 jun 2024	ibRnlCu7JSMHLfUqgrzlrirPKH...
mmontalvo@stackbuild...		24 jun 2024	4 jul 2024	wrbNASPNz4SSGYthTi0iWWu...

Figura 50 Gestión de usuarios con Firebase

Pruebas De Aceptación De Usuario

Mediante un formulario de Google Forms se recopilaron los resultados de las pruebas en el Anexo II. Los resultados se presentan en una tabla que resume el número de tareas completadas, no completadas y las observaciones sobre las tareas que no se completaron en específico para cada caso de prueba.

Tabla 6 Resultados de pruebas de aceptación

Caso de Prueba	Número de Tareas Completadas	Número de Tareas No Completadas	Observaciones
Registro con Google y Carga de Información	14	0	N/A
Registro con Correo Electrónico, Login y Carga de Información	14	0	N/A
Visualización de Pantalla Principal y Componente Vocacional	12	2	Paso 3: Selección de 1 a 5 competencias, Paso 4: Visualización del top 3 de carreras recomendadas

Visualización de Pantalla Principal y Componente de Parejas	13	1	Paso 3: Visualización de persona con mayor compatibilidad
Visualización de Pantalla Principal y Componente Laboral	12	2	Paso 3: Visualización de cargos recomendados

Según Tabla 6 se evidencia que los sistemas de registro y carga de información, tanto a través de Google como de correo electrónico, funcionan correctamente. Sin embargo, se detectaron fallas en la visualización de la pantalla principal y en los componentes vocacionales, de parejas y laborales en casos particulares de usuarios. En el componente vocacional, se identificaron problemas en la selección de competencias y la visualización de carreras recomendadas. El componente de parejas presentó dificultades en la visualización de la persona más compatible, y el componente laboral mostró fallas en la visualización de cargos recomendados. Estas fallas fueron corregidas conforme se detectaron, implementando estrategias con mayores ayudas visuales y guías para reducir estos errores y mejorar la experiencia del usuario.

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

El uso de las metodologías CRISP-DM y Kanban en el desarrollo del proyecto permitió una estructuración efectiva y organizada del trabajo, facilitando la gestión y seguimiento de las tareas. La metodología CRISP-DM proporcionó un enfoque sistemático para la minería de datos, desde la comprensión del negocio hasta la evaluación de los resultados, siguiendo rigurosamente la comprensión de datos, preparación de datos, modelado, evaluación y despliegue, alcanzando una clasificación precisa y generalizable. Por otro lado, la metodología Kanban permitió una gestión ágil del proyecto, adaptándose a los cambios y mejorando la productividad del equipo. La visualización del flujo de trabajo mediante tableros Kanban ayudó a identificar y eliminar cuellos de botella, asegurando una entrega continua y eficiente de las tareas. El análisis inicial del negocio reveló la falta de profesionales en grafología en Ecuador y la necesidad de un sistema que permita la evaluación de personalidades mediante firmas manuscritas, lo que validó la relevancia del proyecto y su potencial impacto en la industria de recursos humanos y terapias individuales.

La metodología CRISP-DM mostró gráficos de precisión y pérdida con una mejora constante durante el entrenamiento y la validación del modelo, donde la aplicación de técnicas como el EarlyStopping y el ajuste de la tasa de aprendizaje mediante ReduceLROnPlateau contribuyeron a evitar el sobreajuste y mejorar la generalización del modelo. Las métricas de evaluación, incluyendo recall, F1-score y la matriz de confusión, indicaron que el modelo alcanzó un nivel aceptable de precisión para clasificar las firmas manuscritas en cuatro temperamentos; sin embargo, las clases de nervioso y sanguíneo mostraron mayores tasas de clasificación correcta, mientras que las clases de bilioso y linfático presentaron mayores desafíos, indicando áreas de mejora en el reconocimiento de características específicas. Por otro lado, la metodología Kanban permitió una gestión eficiente del trabajo en progreso (WIP), asegurando que el equipo no estuviera sobrecargado y que las tareas se completaran de manera oportuna. Los indicadores de rendimiento, como el tiempo de ciclo y la tasa de flujo, demostraron mejoras en la productividad y la entrega continua del proyecto.

El proyecto logró desarrollar un sistema web que permite la clasificación de personalidades a partir de firmas manuscritas utilizando técnicas avanzadas de machine learning. El sistema ofrece una interfaz amigable para los usuarios, donde pueden cargar sus firmas y obtener un análisis detallado de su personalidad. Además, implementar un modelo de redes neuronales convolucionales con transfer learning demostró ser eficaz en la clasificación de personalidades, proporcionando resultados adecuados bajo la aprobación de nuestro grafólogo. Además, el proyecto logró integrar el modelo en una API, facilitando su despliegue y uso en entornos de producción.

4.2 Recomendaciones

Se recomienda continuar recopilando datos de firmas manuscritas para mejorar la precisión y generalización del modelo. La inclusión de más datos etiquetados permitirá un entrenamiento más robusto y una mejor clasificación de las clases que presentan fallos en detección de características particulares. Además, es importante desarrollar nuevas funcionalidades en el sistema web, como la integración con redes sociales y plataformas de citas, para ampliar el alcance y la usabilidad del sistema.

Realizar pruebas con usuarios externos y profesionales en grafología es esencial para validar la efectividad del modelo y del sistema en diferentes contextos y escenarios de uso. También es crucial continuar optimizando la infraestructura de despliegue utilizando tecnologías *serverless* y contenedores para asegurar la escalabilidad y disponibilidad del sistema.

Finalmente, se debe ofrecer formación y capacitación a los usuarios sobre el uso del sistema y la interpretación de los resultados grafológicos. Esto maximizará el valor y el impacto del proyecto en la industria.

5 REFERENCIA BIBLIOGRÁFICA

- [1] "THE LEGAL IMPLICATIONS OF GRAPHOLOGY".
- [2] "Graphology: Definition, Explanation, Purpose & Examples." Accessed: Jul. 11, 2024. [Online]. Available: <https://www.studysmarter.co.uk/explanations/english/graphology/>
- [3] A. : María, F. Centeno Muñoz, C. Marín Hernández Colaborador, and J. T. Molina, "COLEGIO MEXICANO DE GRAFOLOGÍA Y CRIMINALÍSTICA".
- [4] "Grafometrics - Evaluación Psicolaboral online y Detección automatizada del talento." Accessed: Jul. 11, 2024. [Online]. Available: <https://web.grafometrics.com/>
- [5] "Grafología de los temperamentos hipocráticos." Accessed: Jul. 11, 2024. [Online]. Available: <https://sandracerro.com/grafologia-de-los-temperamentos-hipocraticos/>
- [6] "Conceptos básicos de ayuda de CRISP-DM - Documentación de IBM." Accessed: Jul. 11, 2024. [Online]. Available: <https://www.ibm.com/docs/es/spss-modeler/saas?topic=dm-crisp-help-overview>
- [7] "La metodología CRISP-DM en ciencia de datos - IIC." Accessed: Jul. 11, 2024. [Online]. Available: <https://www.iic.uam.es/innovacion/metodologia-crisp-dm-ciencia-de-datos/>
- [8] J. G. Roig, "Metodologías y estándares".
- [9] "CRISP-DM, Metodología de Datos - iPMOGuide." Accessed: Jul. 11, 2024. [Online]. Available: <https://ipmoguide.com/crisp-dm-metodologia-de-datos/>
- [10] "Benefits of Kanban Methodology for Software Development - Cprime." Accessed: Jul. 11, 2024. [Online]. Available: <https://www.cprime.com/resources/blog/benefits-of-kanban-methodology-for-software-development/>
- [11] "¿Qué es la metodología Kanban y cómo funciona? [2024] • Asana." Accessed: Jul. 11, 2024. [Online]. Available: <https://asana.com/es/resources/what-is-kanban>
- [12] "The Kanban system for agile software development explained." Accessed: Jul. 11, 2024. [Online]. Available: <https://kruschecompany.com/kanban-method-agile-software-development/>
- [13] "Manage columns on your board - Azure Boards | Microsoft Learn." Accessed: Jul. 11, 2024. [Online]. Available: <https://learn.microsoft.com/en-us/azure/devops/boards/boards/add-columns?view=azure-devops&tabs=process-administrator>
- [14] "What Is Machine Learning (ML)? | IBM." Accessed: Jul. 11, 2024. [Online]. Available: <https://www.ibm.com/topics/machine-learning>
- [15] "Aprendizaje supervisado frente a aprendizaje no supervisado: diferencia entre los algoritmos de machine learning - AWS." Accessed: Jul. 11, 2024. [Online]. Available: <https://aws.amazon.com/es/compare/the-difference-between-machine-learning-supervised-and-unsupervised/>

- [16] “¿Qué es el aprendizaje profundo? - Explicación del aprendizaje profundo - AWS.” Accessed: Jul. 11, 2024. [Online]. Available: <https://aws.amazon.com/es/what-is/deep-learning/>
- [17] “¿Qué son las redes neuronales convolucionales? | IBM.” Accessed: Jul. 11, 2024. [Online]. Available: <https://www.ibm.com/es-es/topics/convolutional-neural-networks>
- [18] “¿Qué son las redes neuronales convolucionales? | 3 cosas que debe saber - MATLAB & Simulink.” Accessed: Jul. 14, 2024. [Online]. Available: <https://la.mathworks.com/discovery/convolutional-neural-network.html>
- [19] “Cómo Entrenar una Red Neuronal Convolucional (CNN) en CIFAR-10 con Keras - DataSmarts Español.” Accessed: Jul. 11, 2024. [Online]. Available: <https://datasmarts.net/es/como-entrenar-una-red-neuronal-convolucional-cnn-en-cifar-10-con-keras/>
- [20] “Transfer Learning - MATLAB & Simulink.” Accessed: Jul. 11, 2024. [Online]. Available: <https://la.mathworks.com/discovery/transfer-learning.html>
- [21] “Learn web development | MDN.” Accessed: Jul. 14, 2024. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Learn>
- [22] “Why Use React: A powerful tool for web development.” Accessed: Jul. 11, 2024. [Online]. Available: <https://www.angularminds.com/blog/why-use-react-for-web-development>
- [23] “React.” Accessed: Jul. 11, 2024. [Online]. Available: <https://es.react.dev/>
- [24] “Introduction to Nest JS | Refine.” Accessed: Jul. 11, 2024. [Online]. Available: <https://refine.dev/blog/nest-js/>
- [25] “¿Qué es la arquitectura de microservicios? | Google Cloud.” Accessed: Jul. 11, 2024. [Online]. Available: <https://cloud.google.com/learn/what-is-microservices-architecture?hl=es-419>
- [26] “Redis - La plataforma de datos en tiempo real.” Accessed: Jul. 11, 2024. [Online]. Available: <https://redis.io/es/>
- [27] “Jira | Software de seguimiento de proyectos e incidencias | Atlassian.” Accessed: Jul. 11, 2024. [Online]. Available: <https://www.atlassian.com/es/software/jira>
- [28] “Confluence | El espacio de trabajo remoto para tu equipo | Atlassian.” Accessed: Jul. 11, 2024. [Online]. Available: <https://www.atlassian.com/es/software/confluence>
- [29] “What is CRISP DM? - Data Science Process Alliance.” Accessed: Jul. 11, 2024. [Online]. Available: <https://www.datascience-pm.com/crisp-dm-2/>
- [30] “Grafología contra el Fraude (Análisis Forense de Grafismo o Escritura).” Accessed: Jul. 11, 2024. [Online]. Available: https://iniciativasempresariales.com/ficha_curso.php?c=851
- [31] “Grafometrics - Evaluación Psicolaboral online y Detección automatizada del talento.” Accessed: Jul. 11, 2024. [Online]. Available: <https://web.grafometrics.com/>
- [32] “Descubre a las personas aprendiendo Grafología | Intensivo | Udemy.” Accessed: Jul. 11, 2024. [Online]. Available: https://www.udemy.com/course/grafologiacapacitacion/?utm_source=adwords&utm_medium=udemyads&utm_campaign=LongTail_la.ES_cc.LATAM&campaigntype=Search&portfo

lio=LATAM&language=ES&product=Course&test=&audience=DSA&topic=&priority=&utm_content=deal4584&utm_term=._ag_121424002019._ad_515898216206._kw._.de_c._dm._.pl._.ti_aud-2297301418005%3Aadsa-1237025621932._.li_9069516._.pd._.&matchtype=&gad_source=1&couponCode=2021PM25

- [33] G. Para, D. De Fraudes Bancarios, and Y. O. Aplicaciones, "GRAFOLOGÍA PARA DETECCIÓN DE FRAUDES BANCARIOS Y OTRA APLICACIONES".
- [34] "Tinder | Conoce amigos y gente nueva para salir." Accessed: Jul. 11, 2024. [Online]. Available: <https://tinder.com/es>
- [35] "POF o Plenty of Fish: qué es y cómo funciona - Conoce una de las plataformas de citas más populares." Accessed: Jul. 11, 2024. [Online]. Available: <https://www.mundodeportivo.com/uncomo/relaciones/articulo/pof-o-plenty-of-fish-que-es-y-como-funciona-53516.html>
- [36] "Videochat Omegle: ¡Habla con desconocidos!" Accessed: Jul. 11, 2024. [Online]. Available: <https://omegle.cc/es/#>
- [37] "handwritten signatures." Accessed: Jul. 12, 2024. [Online]. Available: <https://www.kaggle.com/datasets/divyanshrai/handwritten-signatures>

6 ANEXOS

ANEXO I. Historias de Usuario de todo el sistema

ANEXO II. Enlace Figma Prototipo y Enlace de Despliegue

ANEXO III. Resultado de pruebas de aceptación Google Forms

ANEXO I

Historia de Usuario 1: [HU1: Registro de Usuario](#)

Link:

<https://graphologicistic.atlassian.net/wiki/external/MDBjOWU0NTU1ZTkyNGRmM2E2YjI1NmJiOGNjYTBmYTM>

HU1: Registro de Usuario



Propiedad de [MARCELA ALEJANDRA MONTALVO LEON](#) ...

Última actualización: Jul 15, 2024

Prioridad: **MEDIA**

Descripción:

Como nuevo usuario, quiero registrarme en el sistema mediante mi cuenta de Google o con mi correo electrónico para poder crear una cuenta.

Criterios de aceptación:

- El sistema permite el registro del usuario mediante Google Accounts y correo electrónico, nombre completo y contraseña.
- Los campos de entrada para el registro validan correctamente los formatos de correo, requisitos de contraseña y campos nulos.
- El usuario registrado con Google puede iniciar sesión inmediatamente sin necesidad de ingresar sus credenciales.
- El usuario registrado con correo electrónico deberá iniciar sesión con su correo y contraseña para ingresar al sistema.

Historia de Usuario 2: [HU2: Creación de perfil de usuario](#)

Link:

<https://graphologic.atlassian.net/wiki/external/NmMxY2E5MTdkZmZkNDk5OWEyNWYyNGJmY2U4NmM0MWU>

HU2: Creación de perfil de usuario



Propiedad de [MARCELA ALEJANDRA MONTALVO LEON](#) ...

Última actualización: Jul 15, 2024



Prioridad: **ALTA**



Descripción:

Como usuario autenticado, quiero ingresar la foto de mi firma manuscrita, fecha de nacimiento y género para completar mis datos de perfil.



Criterios de aceptación:

- El usuario debe poder cargar una imagen de la firma en cualquier formato desde la web.
- El sistema debe visualizar la vista previa del archivo cargado con su nombre.
- El usuario registrado con Google puede iniciar sesión inmediatamente sin necesidad de ingresar sus credenciales.
- El usuario debe ser capaz de escoger su fecha de nacimiento con ayuda de un calendario y su género con opción múltiple

Historia de Usuario 3: [HU3: Visualización de análisis de personalidad](#)

Link:

<https://graphologic.atlassian.net/wiki/external/YTNkZWl0YzI0MTM4NDBIN2JjODVhYzk1OGlyODUyNDE>

HU3: Visualización de análisis de personalidad



Propiedad de [MARCELA ALEJANDRA MONTALVO LEON](#) ...

Última actualización: Jul 15, 2024



Prioridad: **ALTA**



Descripción:

Como usuario autenticado, quiero ver un resumen gráfico en base al análisis grafológico de mi firma para conocer mi tipo de personalidad, sexualidad, competencias y valores.



Criterios de aceptación:

- El usuario puede ver el resumen de su perfil creado.
- El usuario puede ver la firma cargada y su temperamento más compatible.
- El usuario puede ver una descripción completa de su tipo de personalidad.
- El usuario puede ver una descripción completa de su tipo de personalidad.
- El usuario puede ver un gráfico estadístico de la compatibilidad de la firma con cada personalidad

Historia de Usuario 4: [HU4: Visualización de rama académica y competencias](#)

Link:

<https://graphologic.atlassian.net/wiki/external/ODE1YWYxYjZmMjhmNGRmMThjMGYwZWNjNmRiMGY0OWM>

HU4: Visualización de rama académica y competencias



Propiedad de [MARCELA ALEJANDRA MONTALVO LEON](#) ...

Última actualización: Jul 15, 2024



Prioridad: **MEDIA**



Descripción:

Como usuario autenticado, quiero obtener las competencias vinculadas a mi personalidad y su rama vocacional, para obtener un detalle claro de mis habilidades.



Criterios de aceptación:

- Como usuario autenticado, quiero obtener las competencias vinculadas a mi personalidad y su rama vocacional, para obtener un detalle claro de mis habilidades.
- El sistema debe mostrar la rama académica por personalidad asignada con una breve descripción del su alcance.

Historia de Usuario 5: [HU5: Visualización y ajuste de recomendación vocacional](https://graphologic.atlassian.net/wiki/external/NDg1MzZjZDMxM2RkNDk5ZTIhY2ZlODdhMmYxMGUxYjk)

Link:

<https://graphologic.atlassian.net/wiki/external/NDg1MzZjZDMxM2RkNDk5ZTIhY2ZlODdhMmYxMGUxYjk>

HU5: Visualización y ajuste de recomendación vocacional



Propiedad de [MARCELA ALEJANDRA MONTALVO LEON](#) ...

Última actualización: Jul 15, 2024



Prioridad: **ALTA**



Descripción:

Como usuario autenticado, quiero entablar una compatibilidad entre mi perfil y 3 carreras de la rama académica para visualizar su descripción y competencias asociadas.



Criterios de aceptación:

- El usuario debe elegir entre 1 y 5 competencias mostradas según su criterio.
- El sistema debe limitar las competencias seleccionadas por el usuario para mostrar la compatibilidad con las carreras.
- El usuario debe visualizar en cada carrera recomendada una breve descripción y las competencias asociadas.
- El sistema debe mostrar la compatibilidad jerárquica entre las competencias de las carreras y del usuario, estableciendo un top 3 de recomendaciones.

Historia de Usuario 6: [HU6: Visualización de perfil más compatible](#)

Link:

<https://graphologic.atlassian.net/wiki/external/ZDNmNGU1OWU4OGU3NDBIYzg4ZWlzMzBhZTNiYTl5NTI>

HU6: Visualización de perfil más compatible



Propiedad de [MARCELA ALEJANDRA MONTALVO LEON](#) ...
Última actualización: Jul 15, 2024



Prioridad: **ALTA**



Descripción:

Como usuario autenticado, quiero visualizar el perfil del usuario más compatible a mi personalidad y sexualidad con sus características para evaluar si es una buena coincidencia para mí.



Criterios de aceptación:

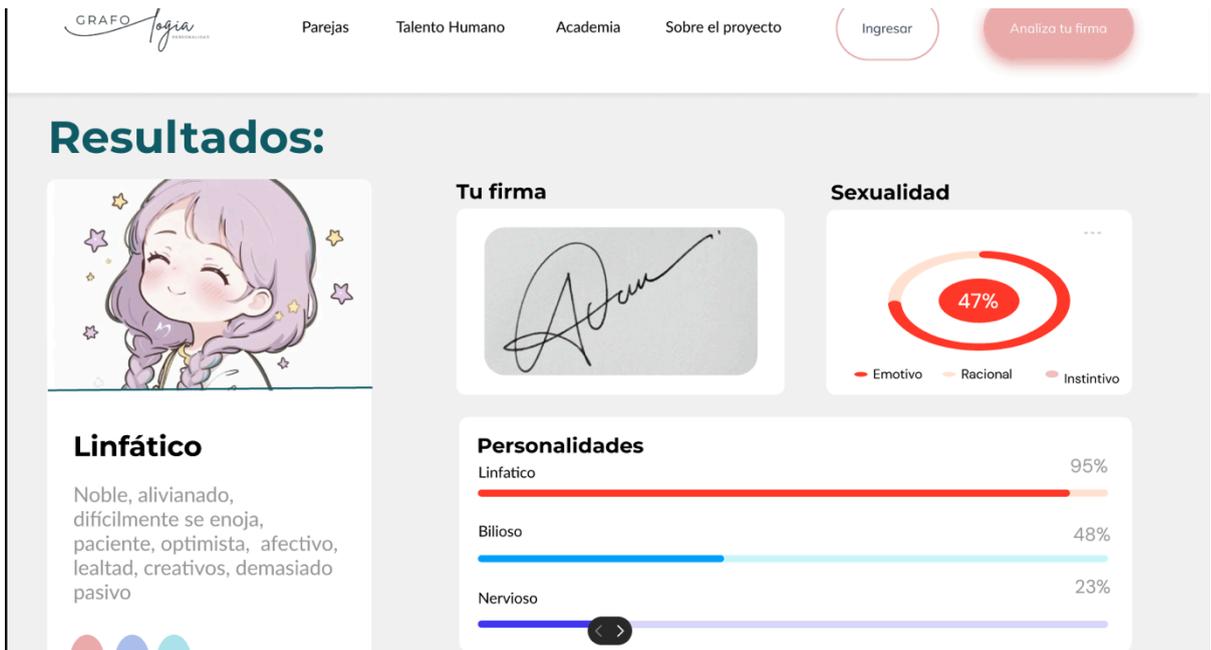
- El usuario puede visualizar el nombre , personalidad y descripción del usuario más compatible según su firma
- El usuario puede visualizar el signo zodiacal y elemento del usuario más compatible
- El usuario puede visualizar la descripción de la sexualidad del usuario más compatible

ANEXO II

Figma Prototipo: [Prototipo Interactivo de Figma](#)

Enlace:

<https://www.figma.com/proto/OTJgOO4hsTInB6BkLldS1v/Graphology?node-id=271-1360&t=1mBjdcBb20kz8EZM-0&scaling=min-zoom&content-scaling=fixed&page-id=251%3A1030&starting-point-node-id=251%3A1031>



Aplicación Desplegada en netlify: <https://graphologist.netlify.app/>



1. Repositorio de Gitlab con código fuente de todos los componentes desarrollados:

Enlace: <https://gitlab.com/KevinR08/tic-graphologist/-/tree/main>

ANEXO III

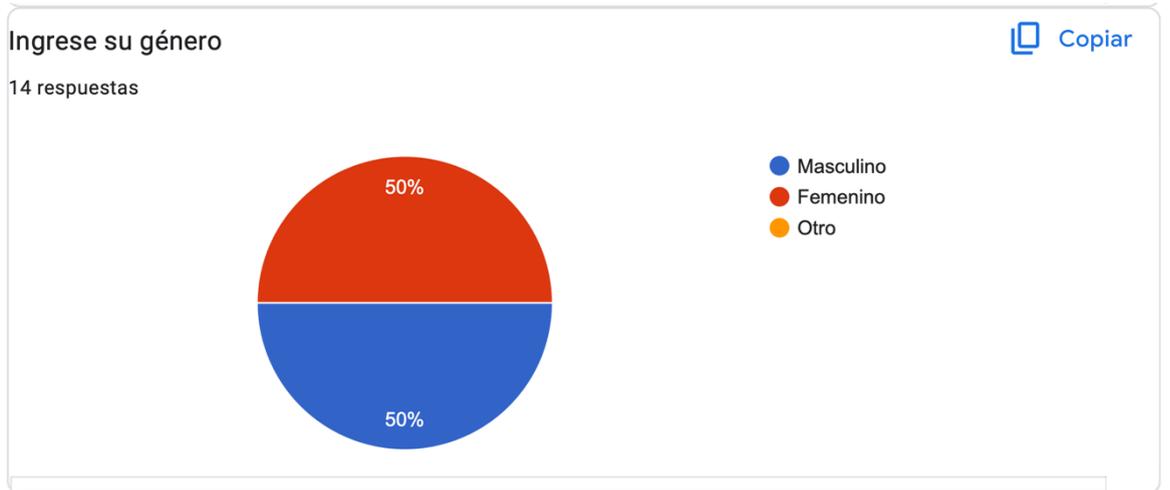
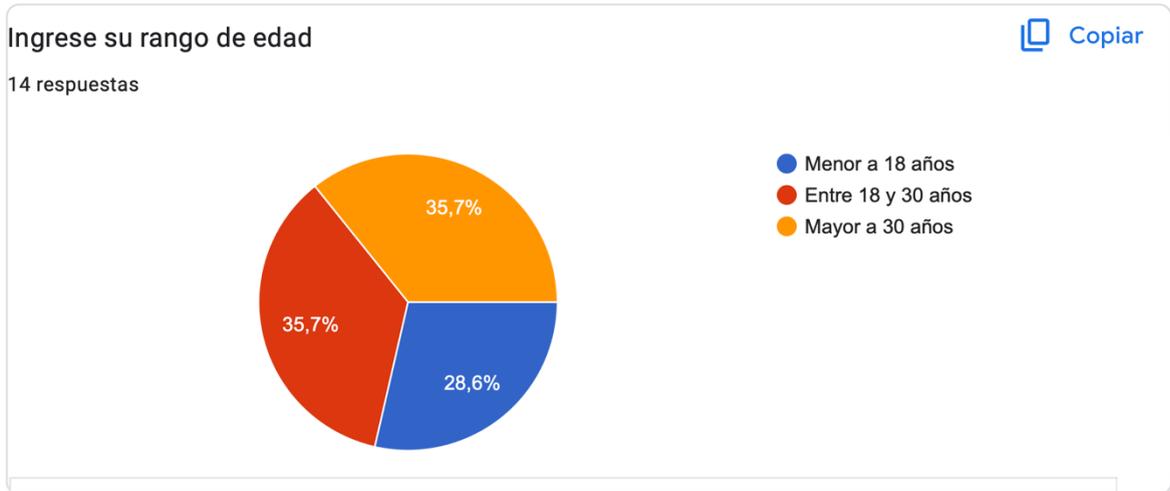
PRUEBAS DE ACEPTACIÓN

14 respuestas

[Ver en Hojas de cálculo](#)

Se aceptan respuestas

[Resumen](#) [Pregunta](#) [Individual](#)

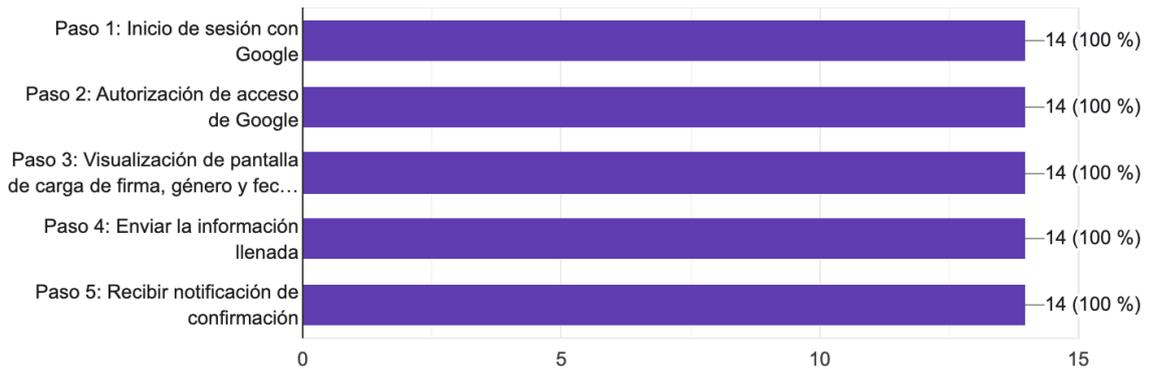


Caso de prueba 1: Registro con Google y Carga de Información.

En el siguiente listado, seleccione las tareas que cumplió con éxito en el caso de prueba 1.

 Copiar

14 respuestas



Caso de prueba 2: Registro con Correo Electrónico, Login y Carga de Información

En el siguiente listado, seleccione las tareas que cumplió con éxito en el caso de prueba 2.

 Copiar

14 respuestas



Caso de prueba 3: Visualización de Pantalla Principal y Componente Vocacional

En el siguiente listado, seleccione las tareas que cumplió con éxito en el caso de prueba 3.

 Copiar

14 respuestas



Caso de prueba 4: Visualización de Pantalla Principal y Componente de parejas

En el siguiente listado, seleccione las tareas que cumplió con éxito en el caso de prueba 4.

 Copiar

14 respuestas



Caso de prueba 5: Visualización de Pantalla Principal y Componente Laboral

En el siguiente listado, seleccione las tareas que cumplió con éxito en el caso de prueba 5.



14 respuestas

