



# INNOVACIÓN y SOFTWARE



Facultad de Ingeniería  
Universidad La Salle, Arequipa, Perú  
facin.innosoft@ulasalle.edu.pe  
<https://revistas.ulasalle.edu.pe/innosoft>



**ARK:** <ark:/42411/s8>



**PURL:** [42411/s8](https://42411/s8)



## Vol. 3 N° 1 2022 Marzo - Agosto

ISSN N°: 2708-0935

DOI: 10.48168/innosoft.s8

ARK: ark:/42411/s8

PURL: 42411/s8

Depósito Legal: 2023-08884

Periodicidad: Semestral

Publicado:30/03/2022

### Editado por:

Universidad La Salle

RUC: 20456344004

Av. Alfonso Ugarte N° 517, Cercado, Arequipa

### COMITÉ EDITORIAL

Editor jefe:

Dr. Yasiel Pérez Vera

Editores asociados:

MSc. Anié Bermudez Peña

MSc. Percy Oscar Huertas Niquén

Miembros del Consejo Editorial

Dr. José Manuel Patricio Quintanilla Paulet

Hno. Jacobo Meza Rodríguez

Dr.C José Javier Zavala Fernández

Dr.C Cristian José López del Álamo

Dr.C Álvaro Rodolfo Fernández del Carpio

MSc. Paul Mauricio Mendoza del Carpio

Corrección de estilos

MSc. Orlando Alonso Mazeyra Guillén

Maquetación

Ronald Fabricio Centeno Cardenas



## EDITORIAL

Prólogo Editorial

**p. 4 - 5**

## ARTÍCULOS ORIGINALES

Aplicación de los árboles de decisión en la identificación de sitios web fraudulentos

Autores: Christian Layme Fernandez, José Manuel Suri Canaza, David Jose Peña Ugarte, Jhon Yoset Luna Quispe

**p. 6 - 16**

Casos de estudio de Design Thinking en las etapas de análisis y diseño del desarrollo de Software

Autores: Victor Fernando Cahui Osis, Darwin Quispe Soto, Andres Condori Huarca, Jesús Chapi Suyo

**p. 17 - 29**

Sistema automático para calificación de vino mediante redes neuronales

Autores: Diego Richard Rivera Demanuel, Cleofe Huamani Huancara, Yimy Alfredo Charca Ccama

**p. 30 - 46**

Revisión de modelos que integren Design Thinking en metodologías de desarrollo ágil

Autores: Dewitt Scott Chavez Ponce, Robert Teodoro Arce Apaza, Andrea Flores Choquehuanca, Daniel Augusto Prado Cussi, Mario Alejandro Huaypuna Cjuno

**p. 47 - 57**

Uso de árboles de decisión para detectar si una habitación está ocupada usando Python

Autores: Joel Atamari Aguilar, Christian Flores Conde, Jhon Mamani Mamani, Sergio Rondon Polanco

**p. 58 - 66**

Design Thinking para resolver problemas con la selección de métricas en la Calidad del Software

Autores: Eduardo Sutta Gonzales, Chirstian Salazar Bobadilla, Julio Aguilar Uñapilco, David Huittocollo Quispe

**p. 67 - 80**

## ARTÍCULOS DE REVISIÓN

Artificial Intelligence in Digital Health: Issues and Dimensions of Ethical Concerns

Autores: Fredrick R. Ishengoma

**p. 81 - 108**



La Revista Innovación y Software de la Facultad de Ingeniería, en la Universidad La Salle, se complace en presentar este primer número de su tercer volumen que tiene como objetivo el promover investigaciones, los cambios y usos de nuevos elementos tecnológicos y su interrelación con la Ingeniería de Software y la Ciencia de la Computación.

Durante la última década, y especialmente en los últimos años, la inteligencia artificial (IA) se ha transformado, no tanto en lo que podemos hacer con ella, sino en lo que hacemos. Durante décadas, incluso antes de que se acuñara el término, la inteligencia artificial despertó miedo e interés cuando los humanos contemplaron la creación de máquinas a su propia imagen. La expectativa de que los artefactos inteligentes deben ser humanoides nos distrae de un hecho importante: hemos estado implementando inteligencia artificial durante algún tiempo.

Esta capacidad de detección de IA se ve amplificada no solo por aumentos masivos en datos digitales y poder de cómputo, sino también por innovaciones en algoritmos de IA. Hoy buscamos fotos, videos y audios. Podemos traducir, transcribir, leer labios, descifrar emociones (incluidas las mentiras), falsificar firmas y otros tipos de escritura a mano y manipular videos. Podemos, y esto es crucial, falsificar audio o video durante transmisiones en vivo, lo que nos permite elegir palabras que millones de personas "presenciarán", especialmente en el caso de celebridades, como los políticos, tenemos una gran cantidad de datos que constituye un modelo exacto.

Pero también hoy, gracias a los avances en robótica, la inteligencia artificial está entrando en nuestros espacios físicos en forma de autos autónomos, armas, drones y dispositivos domésticos, incluidos "altavoces inteligentes" (que en realidad son micrófonos) e incluso consolas de videojuegos. Estamos cada vez más rodeados de conocimientos, análisis y acciones (cada vez más) automatizados y ubicuos, incluso integrados en ellos.

El primer artículo muestra una aplicación de la inteligencia artificial para determinar si un sitio web es fraudulento o no. Se utiliza la técnica de árboles de decisión para establecer, a partir de una serie de parámetros del sitio web, si este es fraudulento o no con cierto grado de probabilidad. La aplicabilidad del modelo presentado permite mejorar técnicas de seguridad informáticas con el fin de hacer el internet un lugar libre de fraudes o estafas usando sitios web que pueden robar o manipular la información de los usuarios.

En el segundo artículo se desarrolla un estudio de cómo la filosofía Design Thinking puede ser usada en las etapas de análisis y diseño durante el proceso de desarrollo de software. En este estudio se presenta una manera óptima de obtener requisitos de software y diseñar arquitecturas orientadas al usuario. Design Thinking genera ideas novedosas en las primeras etapas de desarrollo de software permitiendo que las etapas de implementación y pruebas de un software se lleven a cabo de una mejor manera.



El tercer artículo muestra un modelo de redes neuronales con el objetivo de evaluar la calidad de los vinos en función de sus propiedades. En este trabajo se muestra las diferentes variables que componen un vino y como el modelo propuesto de inteligencia artificial es capaz de evaluar a priori un vino. La aplicación de este modelo en la industria puede ser de gran utilidad ya que se estaría ahorrando tiempo y dinero por los usuarios para conocer la calidad de un vino.

El cuarto artículo se enfoca en un análisis de las metodologías ágiles de desarrollo de software que integran Design Thinking. Se analizan metodologías como el marco de trabajo SCRUM y como este marco de trabajo puede integrar el enfoque de Design Thinking. Se brindan técnicas a incluir en este marco de trabajo para centrarse en la satisfacción del cliente. Se arribó a la conclusión de que SCRUM al inicio de los proyectos no da el suficiente tiempo a sus participantes para asimilar toda la información relevante asociada el proyecto.

En el artículo quinto se muestran un modelo de inteligencia artificial para determinar si una habitación está ocupada o no por personas. En este artículo se utilizan los árboles de decisión para que un sistema experto establezca a partir de varios parámetros como la humedad, la temperatura y el nivel de dióxido de carbono si la habitación esta ocupada o no. Las aplicaciones prácticas de este modelo son varias en el campo de la domótica y los hogares inteligente pudiendo usarse para controlar mediante sensores elementos como luces o termostatos de manera automática.

El sexto artículo propone el uso del enfoque de Design Thinking para evaluar y seleccionar las métricas de calidad de software durante la fase de pruebas. La evaluación y selección de las métricas correctas provee una dirección adecuada en el proceso de pruebas de software. Esto nos indica en que debemos centrarnos para que el proyecto tenga éxito. Las fases de Design Thinking permite seguir un marco estructurado para que las métricas seleccionadas sean las que más se ajusten a las necesidades del proyecto.

Finalmente, el séptimo artículo hace una revisión detallada de las implicaciones éticas que tiene la inteligencia artificial en el sector salud. Se centra en cómo la inteligencia artificial influye en los servicios de atención médica digital. Hace un análisis desde el punto de vista de la ética al respecto del tema, sus principales elementos a tener en cuenta y recomendaciones en el uso de la inteligencia artificial en este campo. Expresa que la tecnología es una herramienta que no debe sustituir al ser humano en el campo de la atención médica.

**Comité Editorial**



# Aplicación de los árboles de decisión en la identificación de sitios web fraudulentos

*Application of decision trees in the identification of fraudulent websites*

6

**Christian Layme Fernández**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [claymef@unsa.edu.pe](mailto:claymef@unsa.edu.pe)

**José Manuel Suri Canaza**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [jsurica@unsa.edu.pe](mailto:jsurica@unsa.edu.pe)

**David Jose Peña Ugarte**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [dpenau@unsa.edu.pe](mailto:dpenau@unsa.edu.pe)

**Jhon Yoset Luna Quispe**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [jlunaq@unsa.edu.pe](mailto:jlunaq@unsa.edu.pe)

 **ARK:** [ark:/42411/s8/a49](https://nbn-resolving.org/urn:nbn:pe:unsa-42411-s8-a49)

 **PURL:** [42411/s8/a49](https://nbn-resolving.org/urn:nbn:pe:unsa-42411-s8-a49)

RECIBIDO 05/09/2021 • ACEPTADO 08/11/2021 • PUBLICADO 30/03/2022

## RESUMEN

La seguridad informática, es un área muy importante en cualquier sistema que tenga conexión a internet, debido a que existen sitios Web fraudulentos que pueden realizar acciones delictivas hacia una persona, organización u otra entidad. Por lo cual es necesario poder detectar qué sitios web son fraudulentos antes de poder ingresar a ella, para ello se desarrolló una implementación mediante Árboles de Decisión con el lenguaje de Python para poder detectar y clasificarlos en Legítimos, Sospechosos y Fraudulentos por medio de 1353 casos que clasifican a los sitios webs.

**Palabras claves:** Árbol de Decisión, Python, Seguridad Informática, Web Sites.

## ABSTRACT

*Computer security is a very important area in any system that has an internet connection, because there are fraudulent websites that can carry out criminal actions towards a person, organization or other entity. Therefore, it is necessary to be able to detect which websites are fraudulent before being able to enter it, for this an implementation was developed through*



*Decision Trees with the Python language to detect and classify them as Legitimate, Suspicious and Fraudulent through 1353 cases that they rank websites.*

**Keywords:** *Decision Tree, Python, Computer Security, Web Sites.*

---

## INTRODUCCIÓN

Uno de los cambios más notorios actualmente en el mundo, aún más por el tiempo de cuarentena, es el uso de Internet y de servicios Web, como son el comercio electrónico, el cual aumentó, por las circunstancias que atraviesa el mundo, Internet se convirtió no solo en un gran apoyo para la sociedad, que utiliza el E-commerce para poderse suministrar las necesidades que tenga, Internet también está siendo utilizado por delincuentes que roban la información de muchas personas y cometiendo actos fraudulentos, el cual genera pérdidas de miles de millones cada año, estos sitios web suelen presentarse como sitios web amigables y de fuentes legítimas de información, productos y servicios en línea.

Los sitios web fraudulentos tienen características, como son: **SFH**, Si el sitio web tiene definido la propiedad action en sus formularios de forma correcta dirigiendo los datos de los formularios a direcciones del mismo sitio web. **PopUp Windows**, Si el sitio tiene ventanas emergentes (PopUp). **SSL Final State**, Si el sitio web utiliza conexión SSL, que la misma presente un estado final válido. **Request URL**, cuando son objetos (imágenes, scripts, hojas de estilos) que son cargadas de otra URL distinta a la del sitio. **URL of Anchor**, cuando los objetos de una página son cargados desde el mismo sitio o desde un subdominio del mismo. **Web Traffic** Si tiene configurado un analizador de tráfico web como el Google Analytics. **URL Length** Cantidad de caracteres de la dirección URL. **Age of Domain**, Cantidad de años que lleva activo el dominio de la URL. **Have IP**, Si tiene o no dirección IP. Las medidas existentes, que disponen los buscadores, han mejorado, pero no son suficientes para detectar sitios web fraudulentos, aún más por el esfuerzo de los que generan este tipo de sitios web para poder eludir estas medidas.

El uso de Árboles de Decisión resulta ser una buena opción para poder detectar los sitios web fraudulentos. Un árbol de decisión es una forma gráfica y analítica de representar todos los eventos (sucesos) que pueden surgir a partir de una decisión asumida en cierto momento. Nos ayudan a tomar la decisión "más acertada", desde un punto de vista probabilístico, ante un abanico de posibles decisiones. Permite desplegar visualmente un problema y organizar el trabajo de cálculos que deben realizarse.

El propósito de este artículo, es el desarrollo un detector de sitios web fraudulentos, a través de inteligencia artificial usando árboles de decisión, con uso librerías para desarrollar el aprendizaje en Python, teniendo 1353 casos en los que se clasifican los sitios web, que serán evaluados y empleando la herramienta Graphviz para visualizar el Árbol decisión resultante.



## Herramientas

### Dataset

Es un conjunto de datos conocido también por el anglicismo dataset, es una colección de datos habitualmente tabulada. En general, y en su versión más simple, un conjunto de datos corresponde a los contenidos de una única tabla de base de datos o una única matriz de datos estadística, donde cada columna de la tabla representa una variable en particular, y cada fila representa a un miembro determinado del conjunto de datos en cuestión [1].

La publicación de los conjuntos de datos usados en un experimento son clave para su reproducibilidad, y cada vez son más las leyes públicas y normas de revistas científicas que obligan a hacerlos públicos, para evitar sesgos. Existen muchas fuentes de datasets como kaggle, el mismo google tiene una plataforma para buscar dataset "Dataset Search" [2].

### Colab

Colaboratory es un proyecto de investigación de Google creado para ayudar a difundir la educación y la investigación sobre Machine Learning. Es un entorno de Jupyter que no requiere configuración para su uso y se ejecuta completamente en la nube. [3]

### Funcionamiento

Colaboratory es gratuita y forma parte de la suite de aplicaciones de Google en la nube. Por ello, para utilizarlo basta con acceder a nuestra cuenta de Google y, o bien entrar directamente al enlace de Google Colab o ir a nuestro Google Drive, pulsar el botón de «Nuevo» y desplegar el menú de «Más» para seleccionar «Colaboratory», lo que creará un nuevo cuaderno (notebook) [4].

### Entorno de Ejecución

Cada celda es independiente, pero todas las celdas en un cuaderno utilizan el mismo kernel. El kernel es el motor de computación que está por debajo y que se encarga de ejecutar nuestro código y devolver el resultado para mostrarlo en la celda.

El estado del kernel persiste durante el tiempo, con lo que, aunque cada celda sea independiente, las variables declaradas en una celda se pueden utilizar en las demás celdas [4].



## Librerías

```
from pandas import Series, DataFrame
import pandas as pd
```

Figura 1. Librería pandas

Pandas es una herramienta de manipulación de datos de alto nivel desarrollada por Wes McKinney. Es construido con el paquete Numpy y su estructura de datos clave es llamada el DataFrame. El DataFrame te permite almacenar y manipular datos tabulados en filas de observaciones y columnas de variables [5].

```
import numpy as np
```

Figura 2. Librería numpy

Numpy es la biblioteca central para la computación científica en Python. Proporciona un objeto de matriz multidimensional de alto rendimiento y herramientas para trabajar con estas matrices.[6].

```
import os
import matplotlib.pyplot as plt
```

Figura 3. Librería matplotlib

Matplotlib es una biblioteca de trazado 2D de Python que produce figuras de calidad de publicación en una variedad de formatos impresos y entornos interactivos en todas las plataformas. matplotlib se puede usar en scripts de Python, el shell de python e ipython, servidores de aplicaciones web y seis juegos de herramientas de interfaz gráfica de usuario. matplotlib intenta hacer las cosas fáciles y las difíciles posibles. Puede generar gráficos, histogramas, espectros de potencia, gráficos de barras, gráficos de error, gráficos de dispersión, etc., con solo unas pocas líneas de código [7].

```
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report
import sklearn.metrics
```

Figura 4. Librería sklearn



Scikit-learn anteriormente scikits.learn y también conocido como sklearn es una biblioteca de machine learning de software libre para el lenguaje de programación Python. [9]

Cuenta con varios algoritmos de clasificación, regresión y agrupamiento que incluyen support-vector machines, random forests, gradient boosting, k-means y DBSCAN, y está diseñado para interoperar con las bibliotecas numéricas y científicas de Python NumPy y SciPy, sus principales algoritmos realizan [8]:

- Classification: Identifica a qué categoría pertenece un objeto
- Regression: Predice un atributo de valor continuo asociado con un objeto.
- Clustering: Agrupación automática de objetos similares en conjuntos.
- Dimensionality reduction: Reduce el número de variables aleatorias a considerar.
- Model selection: Comparara, válida y elige parámetros y modelos.
- Preprocessing: Extrae y normaliza las características.

## Métodos y Metodología computacional

Se usó para el desarrollo de este detector de sitios web fraudulentos usamos el modelo CRISP-DM, la cual comprende de seis fases: análisis del problema, análisis de Datos, preparación de los Datos, modelado, evaluación y explotación [10].

- La fuente de información fue un Dataset.csv, esto fue sacado de internet con datos reales, para poder construir un árbol de decisión utilizando las librerías sklearn, y se hizo uso de las funciones que nos brinda.
- Análisis del problema: Identificamos qué tipos de clasificación tendrán las páginas que se van a probar, viendo que pueden ser tres, y también identificamos la información necesaria para poder hacer esta clasificación.
- Análisis de Datos: con el dataset que obtuvimos, vimos que cantidad de información tiene, y las variables lingüísticas las clasificamos en números para su uso correcto con la librería que se trabajaría.
- Preparación de Datos: Se hizo una limpieza del dataset, con las funciones que nos brinda la libre, como eliminación de campos vacíos, datos irrelevantes.
- Modelado: Seleccionamos la técnica adecuada para poder hacer la clasificación.
- Evaluación: Tuvimos que corroborar efectivamente que el modelo escogido se ajuste a lo que estamos buscando, en este caso poder clasificar los diferentes sitios web.



## RESULTADOS Y DISCUSIÓN

Teniendo en cuenta las fases de la metodología CRISP-DM los siguientes son los resultados de cada una de las etapas:

### Análisis del problema

La Dirección de Seguridad Informática de una universidad pretende desarrollar un sistema informático capaz de detectar si un sitio web es fraudulento.

### Análisis de los datos

Del dataset descargado de internet, después de consultar el número de casos de las tablas de hechos para los nueve eventos en cuestión como se muestra en la Figura 5, se cuenta con un número de 1353 datos por cada categoría. Sin embargo, no todos los eventos comparten las mismas características. Una primera modificación de atributos se realizó para estandarizar aquellas variables lingüísticas dentro del dataset.

El número de eventos modificado es 7, y son los siguientes: SFH, popUpWindows, SSLfinalState, Request URL, URL of Anchor, web Traffic, Have IP y Clasificación del sitio web. Con el fin de obtener los atributos estandarizados para el estudio, se realizó una búsqueda y análisis inicial de errores de los atributos, cambiando la codificación utilizando el rango establecido entre -1, 0 y 1. Basado en la información recepcionada se selecciona Clasificación del sitio web como variable objetivo.

Como resultado se obtuvo el repositorio inicial con los 1353 datos, y sirvió de base para las siguientes etapas de este proceso.



Figura 5. Mapa de burbujas de los rasgos a evaluar

## Preparación de los datos

Se realizó un análisis de la calidad de los datos del dataset, donde se identificó por cada atributo el número de valores distintos, el número de valores nulos, el valor máximo, valor mínimo, media y moda. Como resultado de este proceso, se seleccionaron los 9 eventos, sobre el cual se aplicaron las técnicas de modelado. Todos estos atributos son características propias del tema evaluado.

## Modelado

Con el fin de encontrar o descubrir sitios web fraudulentos, se seleccionó el modelo de clasificación basado en árboles de decisión. La clasificación con árboles de decisión considera clases disjuntas, de forma que el árbol conducirá mediante sus hojas a la predicción diseñada. Después de construido el modelo servirá para determinar en nuevos casos, el tipo de sitio web visitado. Para esta tarea, se escogió como clase el atributo clasificación que determina si el sitio web es fraudulenta, sospechosa, o legítima.



Para evaluar la calidad del modelo, se dividió el repositorio de datos en dos conjuntos: entrenamiento y prueba.

Se empleó el 75% de los datos para el entrenamiento del modelo, mientras que el 25% restante se utilizó para la prueba. Los grupos de datos antes explicados se toman de forma aleatoria en cada iteración. Se calculó el error de muestra parcial del modelo por cada iteración. Por último, se construyó el modelo con todos los datos.

Para la lectura de los resultados de la evaluación se utiliza la matriz de confusión. Se visualiza una matriz de confusión tomada de un ejemplo aleatorio del proyecto. Además de la matriz, se visualiza la precisión del árbol, todo esto se muestra en la Figura 6.

Predicted	-1	0	1	All
True				
-1	149	4	18	171
0	5	16	7	28
1	14	6	120	140
All	168	26	145	339

0.8407079646017699

Figura 6. Matriz de confusión y precisión del árbol

## Evaluación

Analizando los resultados de las pruebas de clasificación con árboles de decisión realizadas con el conjunto de datos, en el cual se almacenan los datos de 1353 casos, se puede observar que el árbol de decisión construido con un factor de entrenamiento del 75%, esto quiere decir 1014 casos tomados aleatoriamente, y un número de casos de evaluación correspondiente al 25% que representa a los 339 casos restantes, con 285 instancias correctamente clasificadas, que



corresponden a un porcentaje de precisión del 84,07% y 54 instancias incorrectamente clasificadas, correspondiente a un porcentaje de error del 15,93%.

El árbol de decisión generado evalúa cada caso buscando que la entropía se iguale a cero. En algunas ramas, los resultados se visualizan en capas superiores, y de esta manera, mientras se aumenta la profundidad del árbol se encuentra mayor cantidad de resultados. El árbol de decisión es de tipo binario, ya que solo muestra dos ramas hijas, algunas concluyendo en hojas de manera temprana.

El algoritmo genera la repartición y categorización de los nodos hijos haciendo una evaluación ponderada, utilizando teoría de árboles de decisión, y brindando un número que sirve de particionador en nodo hijo izquierdo y nodo hijo derecho, así sucesivamente hasta llegar al final. Cada nodo hoja evalúa la partición de cada evento, además de su entropía, mostrando también el número de datos evaluados y mostrando cuantos pertenecen a cada sección de la clasificación (fraudulento=-1, sospechoso=0 y legítimo=1).

## Implementación

Teniendo en cuenta los patrones descubiertos, es importante implementar planes de prevención que permitan disminuir los sitios web fraudulentos. Implementar programas de ciber seguridad que permitan prevenir la infiltración delictiva. De igual manera, se deben redoblar esfuerzos en educación de seguridad cibernética y sanciones más enérgicas contra los infractores.

Este conocimiento descubierto se incorporará al existente y se integrará a los procesos de toma de decisiones de los analizadores de sitios web fraudulentos, proporcionando nuevos panoramas.

## Discusión

Según la matriz de confusión de la Figura 2, el modelo clasifica correctamente al 43,95% de casos fraudulentos, al 4,72% de casos sospechosos y al 35,40% de casos legítimos. Por otra parte, la precisión con la clase real de este modelo es de 0,84 que se considera aceptable (1,0 significa que ha habido precisión absoluta).

Los porcentajes de instancias correctamente clasificados presentados en el árbol como en la matriz de confusión indican que el modelo tiene una precisión buena, por lo tanto, es confiable y eficiente, para clasificar nuevos casos, especialmente fraudulentos y legítimos. Este estudio da resultados aceptables que servirán de base al estudio futuro de reconocimiento de sitios web fraudulentos.



## CONCLUSIONES

Los resultados obtenidos con el modelo de clasificación por árboles de decisión, indican que este es capaz de generar modelos consistentes con la realidad observada y el respaldo teórico, basándose únicamente en los datos que se encuentran almacenados en el dataset.

Los porcentajes de instancias correctamente clasificadas presentados en el árbol como en la matriz de confusión muestran que el modelo tiene una precisión buena y por consiguiente es confiable para clasificar nuevos casos, especialmente de sitios web fraudulentos y legítimos. El factor de confianza sirve de respaldo a los porcentajes antes expuestos.

Como trabajos futuros se plantea utilizar otros clasificadores para comparar estos resultados y mejorar la precisión (por ejemplo, regresión logística), especialmente en los sitios web fraudulentos y aumentar el porcentaje de evaluación de sospechosos; aplicar tareas descriptivas de minería de datos como asociación y agrupación, con el fin de encontrar relaciones y similitudes, según el tipo de sitio web analizado. Dentro de esta investigación no hubo conflicto de intereses.

## REFERENCIAS

- [1] "Find Open Datasets and Machine Learning Projects | Kaggle", Kaggle.com, 2020. [Online]. Available: <https://www.kaggle.com/datasets>. [Accessed: 12 Aug 2020].
- [2] "Conjunto de datos", Es.wikipedia.org, 2020. [Online]. Available: [https://es.wikipedia.org/wiki/Conjunto\\_de\\_datos#cite\\_ref-Editorial\\_1-0](https://es.wikipedia.org/wiki/Conjunto_de_datos#cite_ref-Editorial_1-0). [Accessed: 12-Aug- 2020].
- [3] "Colab Notebooks", Magenta, 2020. [Online]. Available: <https://magenta.tensorflow.org/demos/colab/>. [Accessed: 12- Aug- 2020].
- [4] "Google Colab: Python y Machine Learning en la nube - Adictos al trabajo", Adictos al trabajo, 2020. [Online]. Available: <https://www.adictosaltrabajo.com/2019/06/04/google-colab-python-y-machine-learning-en-la-nube/>. [Accessed: 12- Aug- 2020].
- [5] "Pandas Basics - Learn Python - Free Interactive Python Tutorial", Learnpython.org, 2020. [Online]. Available: <https://www.learnpython.org/es/Pandas%20Basics>. [Accessed: 12-Aug- 2020].



- [6] "Python Numpy Tutorial (with Jupyter and Colab)", Cs231n.github.io, 2020. [Online]. Available: <https://cs231n.github.io/python-numpy-tutorial/#numpy>. [Accessed: 12- Aug- 2020].
- [7] S. Programacion en Castellano, "Introducción a la librería Matplotlib de Python", Programación en Castellano., 2020. [Online]. Available: [https://programacion.net/articulo/introduccion\\_a\\_la\\_libreria\\_matplotlib\\_de\\_python\\_1599](https://programacion.net/articulo/introduccion_a_la_libreria_matplotlib_de_python_1599) . [Accessed: 12- Aug- 2020]
- [8] "scikit-learn: machine learning in Python — scikit-learn 0.23.2 documentation", Scikit-learn.org, 2020. [Online]. Available: <https://scikit-learn.org/stable/>. [Accessed: 12- Aug- 2020].
- [9] Fabian Pedregosa; Gaël Varoquaux; Alexandre Gramfort; Vincent Michel; Bertrand Thirion; Olivier Grisel; Mathieu Blondel; Peter Prettenhofer; Ron Weiss; Vincent Dubourg; Jake Vanderplas; Alexandre Passos; David Cournapeau; Matthieu Perrot; Édouard Duchesnay (2011). "Scikit-learn: Machine Learning in Python". Journal of Machine Learning Research. 12: 2825–2830.
- [10] J. Gallardo, "Metodología para el Desarrollo de Proyectos en Minería de Datos CRISP-DM" oldemarrodriguez.com, para. 2, Aug. 12, 2007. [Online]. Available: [http://www.oldemarrodriguez.com/yahoo\\_site\\_admin/assets/docs/Documento\\_CRISP-DM.2385037](http://www.oldemarrodriguez.com/yahoo_site_admin/assets/docs/Documento_CRISP-DM.2385037). [Accessed Aug. 12, 2020].

# Casos de Estudio de Design Thinking en las etapas de Análisis y Diseño del Desarrollo de Software

## *Case Studies of Design Thinking in the Analysis and Design stages of Software Development*

### **Victor Fernando Cahui Osis**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [vcahui@unsa.edu.pe](mailto:vcahui@unsa.edu.pe)

 <https://orcid.org/0000-0003-1587-7545>

### **Darwin Quispe Soto**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [dquispes@unsa.edu.pe](mailto:dquispes@unsa.edu.pe)

### **Andres Condori Huarca**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [acondorihuar@unsa.edu.pe](mailto:acondorihuar@unsa.edu.pe)

### **Jesús Chapi Suyo**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [jchapis@unsa.edu.pe](mailto:jchapis@unsa.edu.pe)

 **ARK:** [ark:/42411/s8/a50](https://nbn-resolving.org/urn:nbn:pe:unsa-42411-s8-a50)

 **PURL:** [42411/s8/a50](https://nbn-resolving.org/urn:nbn:pe:unsa-42411-s8-a50)

RECIBIDO 28/09/2021 • ACEPTADO 25/10/2021 • PUBLICADO 30/03/2022

## RESUMEN

La aplicación de Design Thinking en las primeras etapas del ciclo de vida del desarrollo del software (análisis y diseño), permiten obtener requisitos de manera óptima y diseñar arquitecturas que soluciones del problema del cliente. Como objetivos de este trabajo, se presentan casos de estudio en donde se aplicó Design Thinking en las etapas de análisis y diseño. Como resultados se obtuvo que las ideas generadas utilizando Design Thinking ofrecieron soluciones novedosas, pero algunas de ellas fueron inciertas. Además, se detalla cómo se desarrollaron los casos y cuáles fueron los beneficios de utilizar Design Thinking. Como resultado final, la aplicación de Design Thinking proporciona un buen punto de partida para el desarrollo de un software.

**Palabras claves:** Análisis, Design Thinking, Diseño, Ingeniería, Software.



## ABSTRACT

*The application of Design Thinking in the early stages of the software development life cycle (analysis and design), allows us to obtain requirements optimally and design architectures that solve the client's problem. As objectives of this work, case studies are presented where design was applied with the analysis and design stages in mind. As a result, it was obtained that the ideas generated using Design Thinking offered novel solutions, but some of them were uncertain. It also details how the cases were developed and the benefits of using Design Thinking were experienced. As a final result, the Design Thinking application provides a good starting point for software development.*

**Keywords:** Analysis, Design Thinking, Design, Engineering, Software.

---

## INTRODUCCIÓN

Existen una gran variedad de investigaciones que han sugerido que Design Thinking (DT) puede tener un impacto positivo en el desarrollo de software, ya que facilita una profunda comprensión de las necesidades del usuario y aumenta la colaboración del equipo.

Design thinking es un concepto que promete una mayor innovación a través de un enfoque de innovación más centrado en el usuario [12]. Para mejorar la comprensión y la solución de problemas, así como la atención hacia el diseño, los investigadores sugieren la integración de Design Thinking con el proceso de desarrollo de software [12]. Según [13], usar ambos enfoques juntos tiene un sentido particular porque Design Thinking proporciona una ruta hacia soluciones creativas y centradas en el ser humano, mientras que los métodos ágiles del desarrollo de software optimizan el proceso y permiten que el equipo se mueva rápidamente.

Teniendo en cuenta este punto de vista, los equipos de software podrían beneficiarse de la aplicación de Design Thinking, a su proceso de desarrollo en cada etapa. Sin embargo, todavía no está claro cómo Design Thinking puede ser incorporado al desarrollo de software en una organización grande y cómo se manifiesta durante el proceso de desarrollo de software, que es algo que se explicara con casos de estudio.



## Metodología computacional

### Design Thinking

Design Thinking (DT) es una manera de ofrecer una solución a un problema. Descomponemos un problema, lo dividimos en partes más pequeñas, las analizamos, pensamos mucho, sin límites, todo lo que podamos y todo lo que se nos ocurra, de manera empática y junto a otros miembros del equipo, entonces estaremos mucho más cerca de encontrar la solución que buscamos.[8]

Design Thinking no solo es aplicados en la gestión de proyectos, sino es multidisciplinario, para la realización de este artículo nos centraremos en el área de las etapas de desarrollo de Ingeniería de Software.

### Design Thinking en Ingeniería de Software

En tiempos recientes, Design Thinking se ha utilizado para mejorar el proceso de desarrollo de software y otras áreas relacionadas a la informática, como por ejemplo: enseñar ingeniería de sistemas en puntos críticos para la seguridad, diseño de aplicaciones móviles, desarrollo de juegos, etc[5]. Sin embargo, según los informes, se ha sugerido que el Design Thinking podría usarse también para el aprendizaje basado en proyectos, la obtención de requisitos y la ciberseguridad. Además, el método ágil podría combinarse con el pensamiento de diseño para mejorar el desarrollo de software [5][1].

Según [3], los investigadores han estado discutiendo cómo facilitar la integración de Design Thinking al desarrollo de software con el fin de integrar conocimientos más diversos para abordar los desafíos apremiantes durante el proceso. Según [4], Design Thinking puede soportar el desarrollo de software de cuatro maneras.



Tabla 1: Posibles modelos de integración de Design Thinking en el proceso de desarrollo de tecnologías de información.

Nombre	Descripción
Modelo de proyecto dividido	Design thinking como un proyecto separado del desarrollo
Modelo de términos superpuestos	Design thinking como fase inicial del proceso con uno o más miembros del equipo de desarrollo que participan
Modelo de proyecto indefinido	Design thinking como fase inicial del proceso y una gran superposición del equipo Design thinking y el equipo de desarrollo
Moldeo toolbox	Los desarrolladores de métodos pueden usar para superar problemas que no pueden resolver con métodos de TI

## Etapas de Design Thinking

Design Thinking se define normalmente desde una perspectiva abstracta de alto nivel, pero vista desde un punto práctico, existen diversas estrategias y formas para implementarlo "de forma práctica"[2]. Las siguientes son las etapas generalmente aceptadas como marco para Design Thinking [5], aunque los pasos y la nomenclatura específicos pueden cambiar según la fuente:

- Inspiración:
  - Empatizar: comprender lo que el usuario o el cliente necesita, en términos de lo que siente el cliente.
  - Definir: Establecer las necesidades del cliente en términos de un problema que se puede resolver.
- Ideación: La propuesta de una solución (producto, servicio, experiencia) que satisfaga las necesidades del cliente.
- Implementación:
  - Prototipo: Crear una solución que cubra una selección de las características requeridas por el usuario, para que el cliente final pueda experimentarlo.
  - Prueba: Utilización real de la solución, de manera que el cliente pueda proporcionar comentarios oportunos y el proceso pueda iterar.



Tabla 2: Etapas de Design Thinking [5].

<b>Design Thinking</b>
<i>Empathize:</i> Market research, interviews, user persona, user journey
<i>Define:</i> Point-of-view statemet, Analysis/synthesis, problem statement
<i>Ideate:</i> Brainstorming, point-of-view analysis, <i>How might we templates</i>
<i>Prototype:</i> Storyboard, sketching, non-functional prototyping, functional prototyping
<i>Testing:</i> Minimum viable product rollout, user feedback grid

Design Thinking propone que a través de la empatía y definición, la experimentación, la creación de prototipos y la retroalimentación inmediata, los diseñadores son más propensos a tener éxito al resolver una necesidad, simplificando y humanizando la solución [6].

Design Thinking, por lo tanto, puede ser lo suficientemente flexible como para ser tomado como un marco general de desarrollo de soluciones que lo hace adecuado para ser utilizado en los procesos de desarrollo de software, que de hecho es un campo en el que se requiere un enfoque abierto, creativo y flexible para resolver problemas y encontrar oportunidades de mejora [1].

## Design Thinking aplicado al Análisis de Requerimientos

### Caso de Estudio [6]

Proyecto Capstone de la Universidad Tecnológica de Lappeenranta, aplicado en el curso Human-Centric SE. El objetivo principal de este artículo es permitir a los estudiantes explorar problemas y estimar posibles soluciones, los estudiantes eligieron un escenario donde su solución se desarrollarían, luego elaboraron un inicial plan de proyecto [6], en total son 6 proyectos a los que se evalúan obteniendo como resultado una mejora en las habilidades blandas, una mejor experiencia de trabajar en un proyecto.



## Caso de Estudio [9]

Este artículo de investigación nos plantea la creciente necesidad de digitalizar todo para adaptarse a las tecnologías de tablets, smartphone, etc. El análisis de requerimientos siempre varía entorno a las necesidades de los usuarios, por lo es necesario poder extraer estos requerimientos de manera más dinámica. El análisis de requerimientos también requiere de procedimientos o metodologías que se adaptan a la ingeniería de software, para esta investigación se utiliza los 4 principios de Design Thinking [8], que aporta para mejorar los procesos creativos, la comprensión, el trabajo colaborativo en equipo y la retroalimentación en el proyecto.

## Design Thinking aplicado al Diseño

Design Thinking es una herramienta poderosa que a través del diseño específico permite a las empresas hacer que los clientes hagan lo que quieran [2].

## Caso de Estudio [7]

¿Por qué los profesionales de la industria de Tecnologías de la información se esfuerzan tanto para que cada detalle en el diseño del producto sea perfecto y piensen en los colores de los botones, etc.?

Para los programadores el color de un botón no es importante, ya que vista desde el punto de vista de desarrollo, es cambiar el valor a una variable, pero en la práctica, este es un factor importante, ya que en cierta medida determina si los clientes utilizarán su producto.

A veces, los usuarios no pueden encontrar instantáneamente un botón que están buscando y luego entran en juego dos escenarios:

1. Molestos, cierran la ventana de la aplicación.
2. Pasan bastante tiempo tratando de encontrar el botón que conduce a la funcionalidad que necesitan.

Ambos casos son malos para el dueño del sistema. Y ambos significan que el problema del usuario no se tuvo en cuenta en la etapa de diseño.

Design Thinking en esta etapa permite a las empresas evitar problemas como este.



- Paso 1 Empatizar: Nos ponemos en el caso de que tu eres la persona que entrara a esta aplicación o página web.
- Paso 2 Definir: Cuando se tiene un conocimiento profundo de cuál es el problema del usuario.
- Paso 3 Idear: Podemos sugerir algunas soluciones a este problema.
- Paso 4 Prototipar: Elegir y crear el mejor diseño posible para todos los elementos del producto.
- Paso 5 Probar: Mostramos a los usuarios el diseño de las interfaces. Como resultado, los usuarios finales no se confunden cuando utilizan su producto y eventualmente se convierten en sus clientes leales.

Los mismos colores de botón pueden alentar a los usuarios a tomar una acción que usted necesita para, por ejemplo, programar un seminario web o solicitar la versión de prueba de su producto.

O simplemente mire las notificaciones prominentes del navegador: cuando, por ejemplo, un sitio web de medios que está visitando le ofrece notificaciones instantáneas de noticias importantes.



Figura 1.

El primero es el botón Sí (permitir), al hacer clic en el que acepta recibir notificaciones push. Otro es el botón No, gracias (bloquear).

Estamos acostumbrados al hecho de que el color verde generalmente implica consentimiento, mientras que su color complementario, rojo, significa la disidencia. Como las señales de tráfico rojas dicen que algo está prohibido, un botón rojo dice que al hacer clic no estamos de acuerdo en recibir notificaciones automáticas del sitio web.

El truco es que algunos medios ponen la etiqueta Sí en el botón rojo. También tenga en cuenta el orden: sí, generalmente va primero, pero no en este caso, haciendo un prototipo:



Figura 2.



Por lo tanto, nosotros, que no queremos empeñarnos en spam, intuitivamente hacemos clic en el botón rojo para rechazar la oferta de recibir notificaciones, pero, en cambio, comenzamos a recibirlas. En la parte de desarrollo de software de Diseño, el color no es el único parámetro que importa. El tamaño, la forma, la ubicación, el orden, las etiquetas de los botones y más son cosas a tener en cuenta, los botones son solo la punta del iceberg

## Resultados y discusión

En [5] se tomó a un grupo de 17 estudiantes de tercer año provenientes de tres países diferentes, con especialización en Ciencias de la Computación. El grupo se organizó en cinco equipos de tres miembros y dos equipos formados por cuatro estudiantes. Luego del desarrollo de las actividades correspondientes al Design thinking se obtuvo lo siguiente:

- **Empatizar:** Los equipos encontraron una necesidad que pueda satisfacerse mediante un sistema de software. Pero con la condición que el equipo tiene que consultar a sus amigos y descubrir cuáles son sus necesidades. Luego de la identificación de la necesidad, se le pidió que identifiquen quién es el usuario que padece esa necesidad.
- **Definir:** una vez que se identifica la necesidad y el grupo objetivo, los estudiantes exploraron más a través de entrevistas para delinear una mejor definición de la situación que quieren resolver. Las metas y los objetivos generalmente se escriben en forma de oraciones inspiradas en historias de usuarios. Una vez concluida esta etapa los estudiantes ya tenían claro cuál es la solución.
- **Idear:** Los equipos proponen soluciones, verifican la viabilidad, comparan tecnologías y presentan una idea para desarrollar. En esta fase, se observó que después de ejercitar la definición del problema y las soluciones de lluvia de ideas, un equipo podría transformar la idea original en otra diferente.
- **Prototipo:** Para la elaboración real de una solución de trabajo, los equipos utilizaron la tecnología de su elección, lo que conduce a una variedad de lenguajes de programación y una pila de tecnología muy diversa.
- **Prueba:** Etapa en la cual los estudiantes mostraron todos sus productos.



Como resultado se obtuvo lo siguiente:

- El equipo que utilizó Design Thinking utilizó una metodología de creatividad e innovación para proporcionar una solución de software. Además, los proyectos resultantes del equipo de Design Thinking resultaron ser más innovadores, pero más inciertos.

En [6] nos demuestra cuáles fueron los resultados de aplicar Design thinking en los cursos Capstone, desde la perspectiva de los alumnos se describe lo siguiente:

- Personas: descubrieron que la creación de clientes en las etapas de análisis y diseño inspira a los estudiantes de ingeniería a ver los problemas desde la perspectiva de un grupo de usuarios, en lugar de desde una perspectiva tecnológica y propia.
- Storyboard: descubrieron que expresar ideas con comportamientos y acciones reales de los usuarios a través de storyboards, ayuda a los estudiantes a comprender mejor quién, qué, cuándo, dónde y cómo se usa la aplicación e identificar las implicaciones sociales y éticas.
- Mapa de usuario: descubrieron que el mapa del usuario ayudó a los estudiantes a mejorar y mejorar su pensamiento de diseño al ayudarlos a comprender las emociones de los usuarios de la aplicación en las primeras etapas. En la sesión de tutoría, los estudiantes señalaron que a través del mapa de viaje del usuario lograron comprender los sentimientos más profundos de los usuarios de la aplicación, que de lo contrario podrían haber sido difíciles.
- Creación de prototipos: descubrieron que los estudiantes encontraron que el diseño del prototipo era muy útil para mejorar su proceso de diseño. Por ejemplo, uno de los equipos señaló durante la sesión de tutoría que recibir comentarios de los usuarios durante el proceso del prototipo fue útil para delinear la interfaz de usuario.

En [6] desde la perspectiva de los profesores resaltaron las siguientes lecciones:

- El diseño del curso fue muy efectivo para motivar a los estudiantes a participar, los propios estudiantes eran los encargados de sus proyectos. Al utilizar Design Thinking en las etapas de diseño aumento la motivación de los estudiantes al percibir que lo que diseñan es lo que realmente quiere el usuario.

En [3] los autores aplicaron una serie de preguntas a una empresa de desarrollo de software dividida en tres secciones. La primera sección tenía como objetivo recopilar información demográfica sobre los participantes.



La segunda sección cubrió la experiencia general del entrevistado con DT. La tercera sección tenía como objetivo recopilar datos sobre el valor de DT para el proceso de desarrollo de software y su implementación.

Tabla 3: Estructura de preguntas realizadas a los trabajadores de una empresa de software sobre DT [3]

<b>Part I – Demographics</b>
Age, name
Purpose and task of the team
Role and background of the interviewee
<b>Part II – Experience with DT</b>
What is your experience with Design Thinking?
What is your view point on Design Thinking?
<b>Part III – DT in software development projects</b>
How does Design Thinking support you and your team?
How is Design Thinking implemented?
What activities are you involved in?
How is your involvement with the user?

En [3] los autores aplicaron el cuestionario mencionado anteriormente y como resultado del análisis de las respuestas obtenidas identificaron tres temas principales y ocho categorías que describen las manifestaciones del Design Thinking dentro de la empresa de software. La Tabla de a continuación muestra los temas y categorías relacionadas.

Tabla 4: Temas y categorías de la manifestación del Design Thinking derivado del análisis temático en [3]

Understanding	Implementation	Connection to Agile
Interpretation	DT enablers	Agile activities
Experience	DT barriers	Agile values and principles
Perceived values and benefits	Goals	
	formats	



La comprensión del tema del Design Thinking describe cómo el Design Thinking se manifiesta dentro de las mentes de los empleados. Este tema está cubierto por las categorías interpretación, experiencia, valores y beneficios percibidos. La interpretación de la categoría explica cómo los empleados describen y ven el Design Thinking. La categoría de experiencia cubre lo que los empleados saben sobre el proceso y las herramientas y cuánto han usado y facilitado el pensamiento de diseño. Finalmente, la categoría de valores y beneficios percibidos describe por qué los empleados usarían el Design Thinking.

El tema de conexión a ágil describe cómo el Design Thinking apoya el desarrollo ágil de software. Este tema está cubierto por las categorías de actividades ágiles y valores y principios ágiles. La categoría de actividades ágiles describe qué actividades concretas en el ciclo de desarrollo ágil pueden ser apoyadas por Design Thinking. La categoría valores y principios ágiles describe qué valores y principios ágiles son compatibles con el uso del pensamiento de diseño.

Este estudio tuvo como objetivo comprender cómo DT puede apoyar el desarrollo de software en una empresa de TI y cómo se manifiesta durante el proceso de desarrollo. El primer hallazgo que surgió fue la correlación entre la comprensión del DT y los roles profesionales. Por ejemplo, la interpretación de DT como un proceso o caja de herramientas fue dominante entre los desarrolladores. Ejecutar fases DT más largas no tenía sentido para ellos ni contaban con el conocimiento necesario para hacerlo. Incluso mencionaron la situación en la que DT y su mentalidad no serían de ninguna ayuda, por ejemplo, al solucionar un bug.

La interpretación de DT como una mentalidad fue dominante entre los diseñadores y gerentes. Curiosamente, varios entrevistados se mostraron reacios a la idea de considerar DT solo como un proceso, en lugar de una mentalidad. Sin embargo, no se encontró ningún estudio que afecte negativamente al uso de DT solo como herramienta o proceso. Por el contrario, las investigaciones existentes junto con los hallazgos de este trabajo sugieren que el nivel de aplicación de DT depende del objetivo del proyecto y la experiencia del equipo [10], [11].

## CONCLUSIONES

- Los métodos de Design Thinking son métodos prácticos para involucrar al equipo de desarrollo de software y mejorar su experiencia y habilidades blandas (por ejemplo, el trabajo en equipo) importantes en la industria.
- Cada vez que esté planeando un nuevo producto o un nuevo diseño, aparte de Design Thinking, podemos usar las pruebas A / B, que también se llaman pruebas divididas. Lo mismo se aplica al desarrollo de nuevas características.



- Design thinking es un enfoque para crear productos de software personalizados. Reúne los objetivos comerciales de los clientes y las necesidades del usuario final y los integra en una solución perfectamente adaptada que aborda el problema de una manera efectiva.
- Design thinking mejora el proceso de entendimiento por parte de los desarrolladores hacia los usuarios, esto es porque se busca empatizar con el usuario final.
- En la Universidad Tecnológica de Lappeenranta se aplica Design Thinking a seis proyectos distintos, y se pudo obtener resultados positivos, los que más resaltan es la mejor productividad y la disminución de abandonos por parte de los estudiantes.
- Los proyectos realizados utilizando Design Thinking pueden resultar muy innovadores pero pueden ser inciertos.
- El uso de Storyboard y mapas de usuario ayudan considerablemente a entender que es lo que necesita el usuario final.
- La aplicación Design Thinking en una empresa de TI da a conocer la correlación entre la comprensión del DT y los roles profesionales. Es decir, no es lo mismo Design Thinking para un gerente que para un desarrollador, cada uno lo ve y lo aplica desde su perspectiva.
- No se puede generalizar o estandarizar Design Thinking, ya que este es adaptable al entorno en donde se vaya a aplicar.

## REFERENCIAS

- [1] K. Thoring, M. Mülle, Understanding Design Thinking: a Process Model Based on Method Engineering. International Conference On Engineering And Product Design Education 8 & 9 September 2011, City University, London, UK
- [2] J. Denning, The Profession of IT Design Thinking, Communications of the ACM, December 2013
- [3] F. Dobrigkeit, D. de Paula Design, Thinking in Practice: Understanding Manifestations of Design Thinking in Software Engineering, Hasso Plattner Institute Potsdam, Germany. ESEC/FSE '19, Tallinn, Estonia, August 2019
- [4] T. Lindberg, E. Köppen, I. Rauth, C. Meinel, On the perception, adoption and implementation of Design Thinking in the IT industry.



- [5] L. Corral, I. Fronza, Design Thinking and Agile Practices for Software Engineering: An Opportunity for Innovation.
- [6] M. Palacin-Silva, J. Khakurel, Infusing Design Thinking into a Software Engineering Capstone Course. 2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)
- [7] J. Sumato, What Design Thinking Is and How It Is Used in Software Development. Recuperado 23 octubre, 2019, de <https://medium.com/@sumatosoft/what-design-thinking-is-and-how-it-is-used-in-software-development-37cf8c581675>
- [8] T. Brown, Design thinking, Harvard business review, 2008, vol. 86, no 6, pp. 84.
- [9] A. Carrel, A. Lauenroth, Using Design Thinking for requirements engineering in the context of digitalization and digital transformation: a motivation and an experience report. The Essence of Software Engineering. Springer, Cham, 2018. pp. 107-120.
- [10] Franziska Dobrigkeit, Danielly de Paula, and Matthias Uflacker. 2018. InnoDev A Software Development Methodology Integrating Design Thinking, Scrum and Lean Startup. In Design Thinking - Research Looking Further: Design Thinking Beyond Solution-Fixation, Hasso Plattner, Christoph Meinel, and Larry Leifer (Eds.). Springer, 199–228.
- [11] H. Ximenes, I. Alves, and C. Araújo. 2015. Software Project Management Combining Agile, Lean Startup and Design Thinking. In Design, User Experience, and Usability: Design Discourse, Aaron Marcus (Ed.). Vol. 9186. Springer International Publishing, Cham, 356–367. [http://link.springer.com/10.1007/978-3-319-20886-2\\_34](http://link.springer.com/10.1007/978-3-319-20886-2_34)
- [12] T. Brown. 2008. Design thinking. Harvard Business Review 86, 6 (2008), 84. <http://www.unusuallleading.com/wp-content/uploads/2009/12/HBR-on-Design-Thinking>
- [13] W. Brenner, F. Uebernickel, T. Abrell.. Design Thinking as Mindset, Process, and Toolbox. In Design Thinking for Innovation, Walter Brenner and Falk Uebernickel (Eds.). Springer International Publishing, Cham, 3–21, 2016



# Sistema automático para calificación de vino mediante Redes Neuronales.

30

*Automatic system for wine qualification through Neural Networks*

**Diego Richard Rivera Demanuel**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [driverad@unsa.edu.pe](mailto:driverad@unsa.edu.pe)

**Cleofe Huamani Huancara**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [chuamanihu@unsa.edu.pe](mailto:chuamanihu@unsa.edu.pe)

**Yimy Alfredo Charca Ccama**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [ycharca@unsa.edu.pe](mailto:ycharca@unsa.edu.pe)

 **ARK:** [ark:/42411/s8/a51](https://nbn-resolving.org/ark:/42411/s8/a51)

 **PURL:** [42411/s8/a51](https://nbn-resolving.org/ark:/42411/s8/a51)

RECIBIDO 15/10/2021 • ACEPTADO 27/11/2021 • PUBLICADO 30/03/2022

## RESUMEN

Tratamiento de datos para la calificación de vinos, este informe detalla el proceso seguido, en donde se utilizó el lenguaje de programación Python, para el análisis de los datos del dataset, se utilizó el servidor Google Colab para ejecutar los algoritmos en la nube ya que el equipo considero que la velocidad de análisis de datos en google colab es más rápido. Las redes neuronales tienen capacidad de aprender y realizar tareas basadas en un entrenamiento inicial llamado aprendizaje adaptativo y además de que son tolerantes a los fallos.

**Palabras claves:** Redes neuronales, Tratamiento de datos, Datos masivos.

## ABSTRACT

*Treatment of data for the qualification of wines, this report details the process followed, where the Python programming language was used, for the analysis of the data of the dataset, the Google Colab server was used to execute the algorithms in the cloud since the team considered that the speed of data analysis in Google Collab is faster. Neural networks have the ability to learn and perform tasks based on an initial training called adaptive learning and are also fault-tolerant.*

**Keywords:** Neural networks, Data processing, Big data



## INTRODUCCIÓN

Tratamiento de datos es un término utilizado cuando se trata de analizar datos en su mayoría masivos, existen muchos lenguajes de programación con librerías para el tratamiento de datos, y técnicas a utilizar, en este trabajo se presenta un informe sobre un sistema de calificación de vinos para una empresa productora de vino que desea un sistema informático que le permita calificar los vinos que esta produce, se consideró también de que los errores del sistema son mínimos considerando de que el porcentaje de errores es casi nula, en [13] recomienda el uso de la arquitectura de redes neuronales de la Inteligencia artificial donde se emplea el uso del lenguaje de programación Python y Jupyter Notebook como IDE. Se evaluaron los datos del Dataset proporcionado por el docente de la asignatura, el sistema permite al usuario poder definir una cantidad aleatoria de los casos del dataset para poder usarlos en el entrenamiento y definir otra cantidad aleatoria para poder utilizarlos en el entrenamiento, también se tomó en cuenta de que los casos utilizados en el entrenamiento no podrán ser utilizados en la validación del entrenamiento para luego mostrar las medidas de los errores.

El desarrollo del sistema requiere análisis de datos masivos, donde es necesario transformar la dataset, seleccionar un lenguaje de programación y buscar las librerías que trabajan con las redes neuronales, así lo afirma [14].

## Materiales y métodos o Metodología computacional

En este trabajo de investigación se utilizaron los datos de una empresa productora de vino, la dataset fue proporcionado por el docente del curso de Inteligencia Artificial, las principales herramientas son el lenguaje de programación Python con los respectivos paquetes y Jupyter notebook como IDE principal, se utilizó redes neuronales como metodología computacional, consiste en un conjunto de unidades a las que se le llaman neuronas artificiales que van conectadas entre sí para transmitir las señales. En [12] afirma que en esta metodología la información que se utiliza como entrada atraviesa la red neuronal para ser sometida a diversas operaciones que tienen como fin producir valores de salida es así que el Sistema Automático califica los vinos que la empresa produce.

Adicionalmente a esto se utilizó Google Colab que es un entorno portátil de Jupyter que tiene un uso gratuito y se ejecuta completamente en la nube, una característica resaltante de Google Colab es que no requiere configuración y los miembros del equipo pueden editar simultáneamente los cuadernos que se crean, además admite muchas bibliotecas de aprendizaje automático populares que se pueden cargar fácilmente en su computadora portátil, Google Colab se utilizó para pruebas de entrenamiento, optimización de subparámetros y ejecuciones.



Decidimos utilizar la plataforma de Google Colab porque consideramos que el análisis de datos nos tomaría más tiempo en nuestras computadoras personales.

También se escogió la arquitectura del tipo Perceptrón Multicapa que evoluciona del perceptrón simple en donde incorpora capas de neuronas ocultas, con esto consigue representar funciones no lineales, además está compuesto por una capa de entrada, una capa de salida y n capas ocultas entremedias. En [15] afirma que el perceptrón multicapa tiene salidas disjuntas pero relacionadas entre sí, de tal manera que la salida de una neurona es la entrada de la siguiente.

## **Descripción general de la aplicación.**

Una empresa productora de vino desea hacer un sistema informático que le permita realizar la calificación de los vinos que esta produce. En dicho sistema desean utilizar inteligencia artificial para tal efecto. Para esto cuentan con una base de calificaciones históricas hechas por expertos con un total de 6 497 casos. Cada caso cuenta con los siguientes atributos: tipo de vino (blanco o tinto), nivel fijo de la acidez, nivel variable de la acidez, nivel de ácido cítrico, nivel de azúcar residual, cantidad de cloruros, nivel libre de dióxido de azufre, nivel de dióxido de azufre, densidad, pH, nivel de sulfatos, grado de alcohol y la calificación del vino (1 a 10).

Se desea que el sistema sea capaz de calificar un vino en la escala de 1 a 10 dando los valores de los atributos antes mencionados. Para esta tarea utilice el 75% la base de calificaciones anteriores para entrenar el modelo y el 25% de lecturas restantes para comprobar el entrenamiento de forma aleatoria; tenga en cuenta que estos parámetros (el porcentaje de entrenamiento y el de validación del mismo) son suministrados por el usuario. El sistema puede equivocarse como máximo 1 vez entre 10000 veces y el tiempo con que se cuenta para dicho entrenamiento es ilimitado. También debe ser posible guardar el modelo una vez entrenado en un fichero y poder cargarlo en otro momento. El usuario podrá cargar el fichero donde se encuentran las calificaciones históricas, definir los parámetros del entrenamiento y visualizar el error una vez ya entrenado el modelo. También se le podrá suministrar un nuevo conjunto de atributos al modelo ya entrenado y este debe ser capaz de calificar el vino.

## **Selección y justificación del modelo.**

Para la selección del modelo con cual se desarrolló el sistema, evaluamos los aspectos de los modelos aprendidos en clase, en donde resaltamos al modelo de redes neuronales como metodología computacional que consiste en un conjunto de unidades a las que se le llaman neuronas artificiales que van conectadas entre sí para transmitir las señales.



En esta metodología la información que se utiliza como entrada atraviesa la red neuronal para ser sometida a diversas operaciones que tienen como fin producir valores de salida. En [12] menciona que las redes neuronales tienen capacidad de aprender y realizar tareas basadas en un entrenamiento inicial llamado aprendizaje adaptativo. De esta forma, la máquina puede aprender a llevar a cabo ciertas tareas mediante el entrenamiento. Las redes neuronales permiten organizar por sí mismas lo aprendido, mientras que el aprendizaje es la modificación de cada elemento procesado, la auto organización consiste en la modificación de la red neuronal completa para llevar a cabo un objetivo específico, [14] lo especifica.

Además, las redes neuronales tienen la capacidad de ser tolerantes a fallos los primeros métodos computacionales con esta capacidad inherente fueron las redes neuronales con esta característica si se produce un fallo en un número no muy grande de neuronas, el sistema no sufre una caída repentina, aunque el comportamiento si se ve influenciado.

Otra de las prioridades principales de las redes neuronales es que pueden operar en tiempo real, las redes neuronales trabajan mediante conexiones en paralelo, lo que permite grandes velocidades de transmisión y respuesta instantáneas. Por último, una de las ventajas que destaca es la facilidad de inserción en la tecnología existente.

## Lenguaje de Programación y herramientas

### Python

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.

Python tiene una sintaxis sencilla que cuenta con una vasta biblioteca de herramientas, que hacen de Python un lenguaje de programación único, lo cual daría una enorme ventaja al desarrollador y hacen un aliado perfecto para la Inteligencia Artificial. [1]

Permite plasmar ideas complejas con unas pocas líneas de código, lo que no es posible con otros lenguajes. Existen bibliotecas como «Keras» y «TensorFlow», que contienen mucha información sobre las funcionalidades del aprendizaje automático.



## ¿Por qué Python?

Python es un lenguaje que todo el mundo debería conocer. En [16] menciona su sintaxis simple, clara y sencilla; el tipado dinámico, el gestor de memoria, la gran cantidad de librerías disponibles y la potencia del lenguaje y algunas características más hacen que desarrollar una aplicación en Python sea sencillo y muy.

La sintaxis de Python es tan sencilla y cercana al lenguaje natural que los programas elaborados en Python parecen pseudocódigo. Por este motivo se dice que es uno de los mejores lenguajes para comenzar a programar. Algunos casos de éxito en el uso de Python son Google, Yahoo, la NASA, Industrias Light & Magic, y todas las distribuciones Linux, en las que Python cada vez representa un tanto por ciento mayor de los programas con disponibilidad.

## Anaconda

Anaconda es una Suite de código abierto que abarca una serie de aplicaciones, librerías y conceptos diseñados para el desarrollo de la Ciencia de datos con Python. [7]

Características:

- Anaconda Navigator es una interfaz gráfica de usuario GUI bastante sencilla, pero con un potencial enorme.
- Puede gestionar de manera avanzada paquetes relacionados a la Ciencia de datos con Python desde la terminal.
- Permite compilar Python en código de máquina para una ejecución rápida.

## Jupyter notebook

Es una aplicación web de código abierto, desarrollada utilizando lenguaje HTML agnóstico que permite crear, compartir y editar documentos, en los que se puede ejecutar código python, hacer anotaciones, insertar ecuaciones, visualizar resultados y documentar funcionalidades. Entre sus usos están: la limpieza y transformación de datos, la simulación numérica, el modelado estadístico, el aprendizaje automático, etc. [11]



## Librerías y framework

Para el desarrollo de la aplicación se procedió a la instalación de librerías correspondientes a la herramienta elegida, el cual facilitará el tratamiento de los datos, y por ende el entrenamiento de este mismo.

### Keras

En [17] nos dice que el mundo de las redes neuronales está en auge. El hecho de simular el cerebro humano en un ordenador, parece ser una de las tecnologías más prometedoras de la informática, pero aún no se ha conseguido, aunque mediante algoritmos de machine learning, ya es posible entrenar máquinas para que aprendan de forma parecida a como lo haría nuestro cerebro.

Keras es un framework de alto nivel para el aprendizaje, escrito en Python y capaz de correr sobre los frameworks TensorFlow, CNTK, o Theano. Fue desarrollado con el objeto de facilitar un proceso de experimentación rápida. [4]

- Para instalar Keras desde la terminal de anaconda, ejecutamos el siguiente comando, *"conda install keras"*.

### Pandas

En [19] no dice que Pandas es un paquete de Python que proporciona estructuras de datos rápidas, flexibles y expresivas diseñadas para hacer que el trabajo con datos "relacionales" o "etiquetados" sea fácil e intuitivo. Pretende ser el elemento fundamental de alto nivel para realizar análisis de datos prácticos y del mundo real en Python, además de que proporciona estructuras de datos similares a los dataframes de R. Pandas depende de Numpy, la librería que añade un potente tipo matricial a Python. Los principales tipos de datos que pueden representarse con pandas son:

Datos tabulares con columnas de tipo heterogéneo con etiquetas en columnas y filas y Series temporales. [9]

- Para instalar Pandas desde la terminal de anaconda, ejecutamos el siguiente comando, *"conda install pandas"*.



## Numpy

NumPy es una extensión de Python, es de código abierto, que le agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices. [8]

Es una biblioteca de Python para trabajar con arreglos multidimensionales lo que hace que el principal tipo de dato es el arreglo o array, también nos permite trabajar con la semántica de matrices y nos ofrece muchas funciones útiles para el procesamiento de números. [20]

- Para instalar Numpy desde la terminal de anaconda, ejecutamos el siguiente comando, *"conda install numpy"*.

## Matplotlib

Matplotlib es una biblioteca para la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python y su extensión matemática NumPy. Proporciona una API, pylab, diseñada para recordar a la de MATLAB.[10].

Matplotlib.pyplot es una colección de funciones estilo comandos que hacen que matplotlib sea similar a matlab. Cada función de pyplot hace algunos cambios a la figura, funciones como crear figura, crear un área de impresión en una figura, imprimir líneas en el área de impresión, agregar etiquetas a las figuras, etc. [21]

- Para instalar Matplotlib desde la terminal de anaconda, ejecutamos el siguiente comando, *"conda install matplotlib"*.

## Descripción de los procesos

Para tener una red neuronal artificial que funcione según la aplicación que se desee, se debe realizar antes de nada un aprendizaje con unos patrones conocidos. Por lo que para utilizar una red neuronal artificial, el primer paso es el entrenamiento de la misma. En él, se presentan un conjunto de entradas y salidas conocidas, que se propagan a través de una red hasta poder obtener unas ciertas salidas (feedforward), para que después se puedan modificar los pesos y bias de manera que la salida coincida con la deseada (backpropagation).



El segundo paso trata de verificar la red mediante un conjunto de entradas y salidas también conocidas, y de la misma forma haciendo feedforward, pero sin necesidad de backpropagation, dado que en principio la red ya está entrenada correctamente. [2]

Por lo que para completar estos pasos ya mencionados se deben de seguir una serie de procesos.

## Tratamiento de datos.

El procesamiento de datos es parte fundamental de cualquier proceso de aprendizaje automático. En algunos casos, busca corregir deficiencias en los datos que puedan dañar el aprendizaje, como omisiones, ruido y valores extremos. En otros casos, se persigue adaptar los datos al modelo que se pretende entrenar para simplificar u optimizar el proceso. [3]

En la dataset proporcionado, se tenían campos anómalos que a simple vista se podía ver, sin embargo, para el tratamiento de estos se tuvo que elegir diferentes métodos, de acuerdo a la herramienta elegida, dado que son más de 6000 datos. Por lo que se procedió a la transformación de estos.

## Transformación de datos

Para este apartado se seguirá una secuencia de pasos:

- El primer paso que haremos será identificar las variables dependientes e independientes.

## Variables Dependientes:

x\_0: Tipo  
x\_1: Acidez fija  
x\_2: Acidez variable  
x\_3: Ácido cítrico  
x\_4: Azúcar residual  
x\_5: Cloruros  
x\_6: Dióxido de azufre libre  
x\_7: Nivel de dióxido de azufre  
x\_8: Densidad  
x\_9: pH  
x\_10: Sulfatos  
x\_11: Grado de alcohol



## Variables Independientes:

y : Calidad

- El segundo paso que haremos es: identificar nuestros campos o variables categóricas y la existencia de valores outliers en dichos campos.

En la dataset que se nos proporcionó, se identificó sólo un campo categórico, "Tipo", por lo que se le aplicó el preprocesamiento, convertir variables categóricas en variables numéricas.

También se identificaron 3 campos con valores outliers, "Densidad", "Cloruros", "Acidez variable", por lo que de la misma forma se realizó el preprocesamiento.

Para ver la distribución de valores de dichos campos, se generó un histograma de cada uno de ellos. Esto con la función hist(), después se generó el diagrama de caja y bigotes, esto para detectar los outliers.

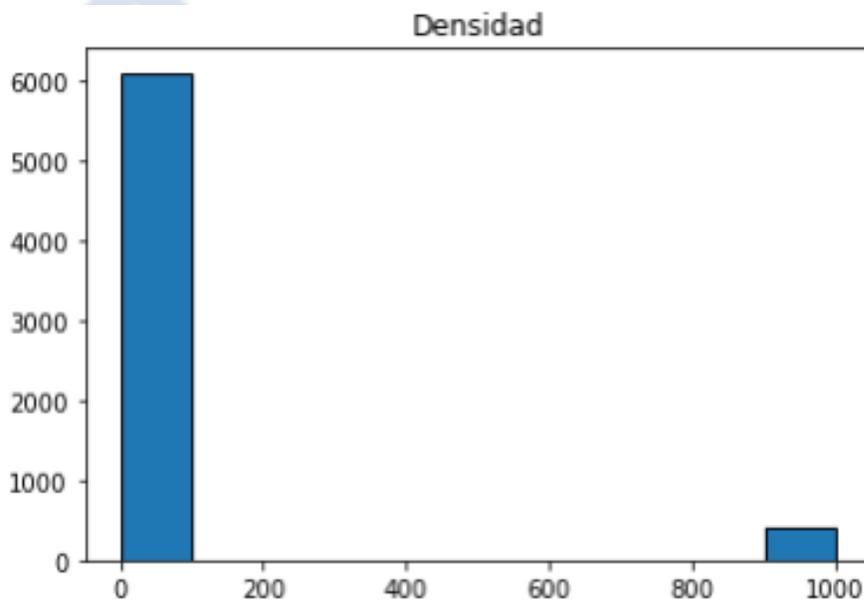


Figura 1. Histograma Densidad

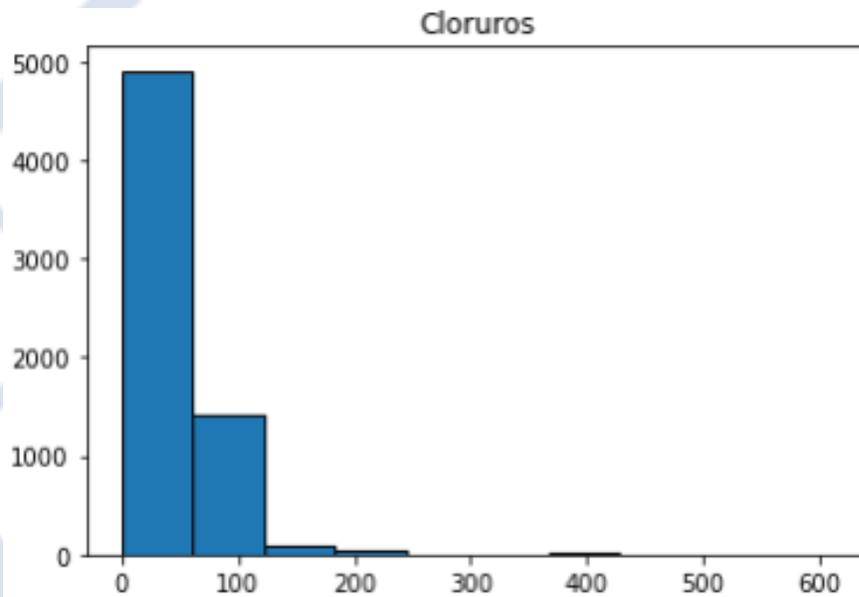


Figura 2. Histograma Cloruros

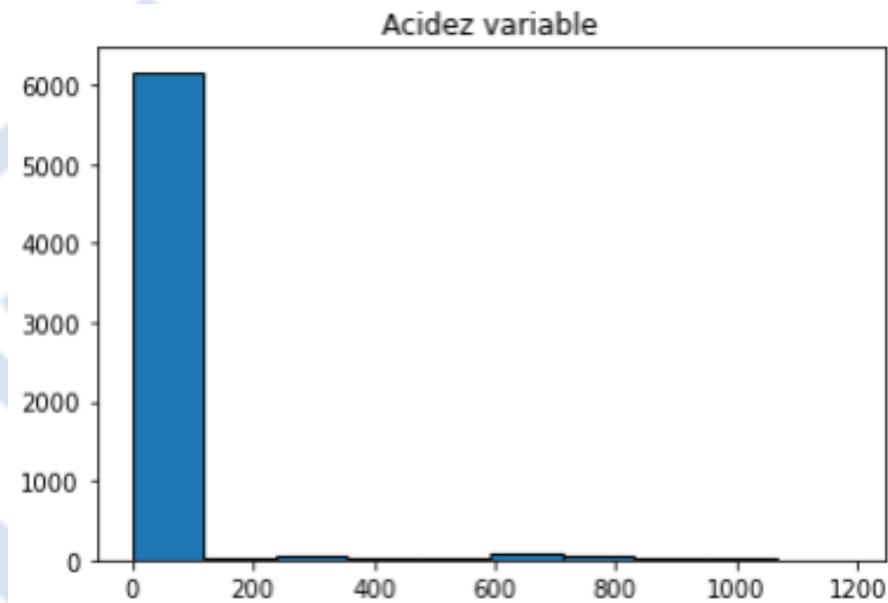


Figura 3. Histograma Acidez variable

En la figura 1, 2 y 3; se puede observar una cantidad de datos atípicos.

Para la identificación de outliers en la dataset, utilizamos el gráfico de caja y bigotes. el cual se formará con la función `boxplot()` del paquete `matplotlib`.

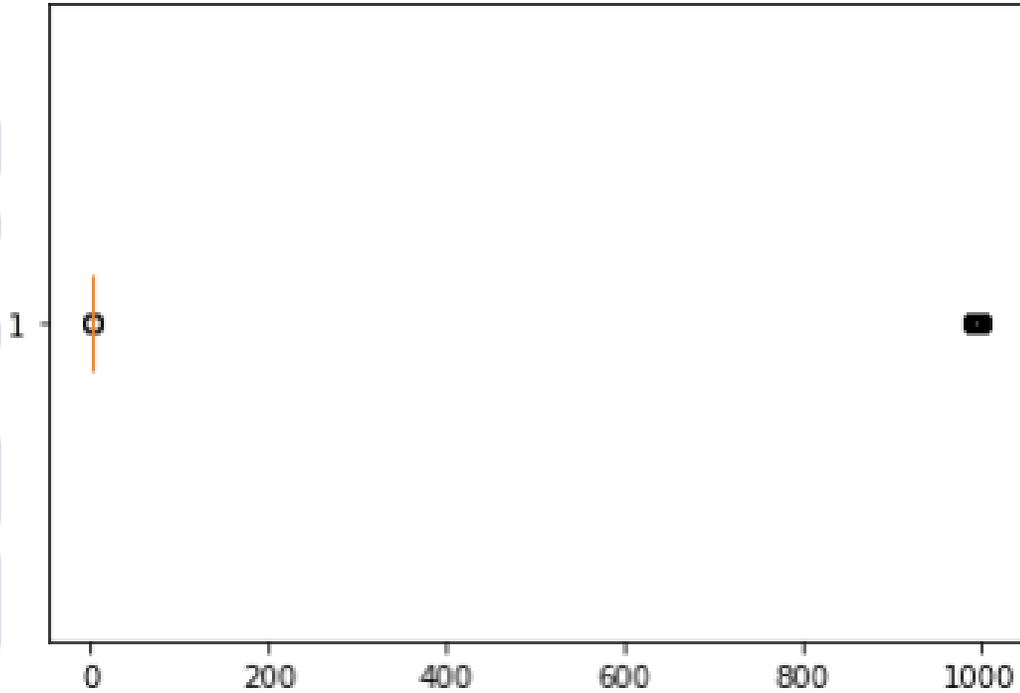


Figura 4. Diagrama de Caja y Bigotes Densidad.

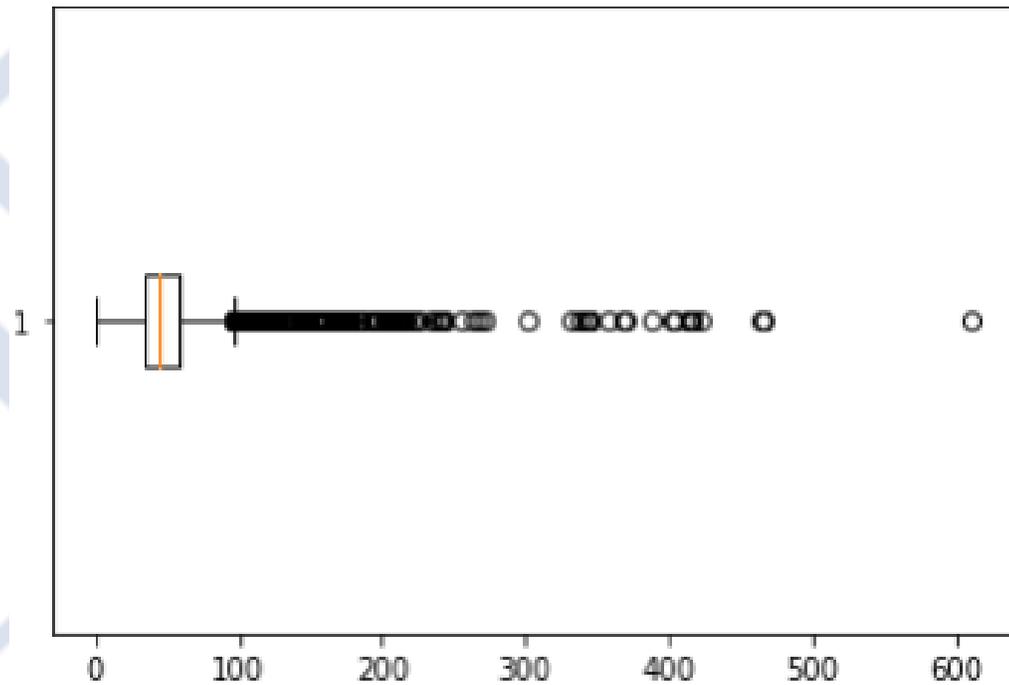


Figura 5. Diagrama de Caja y Bigotes Cloruro.

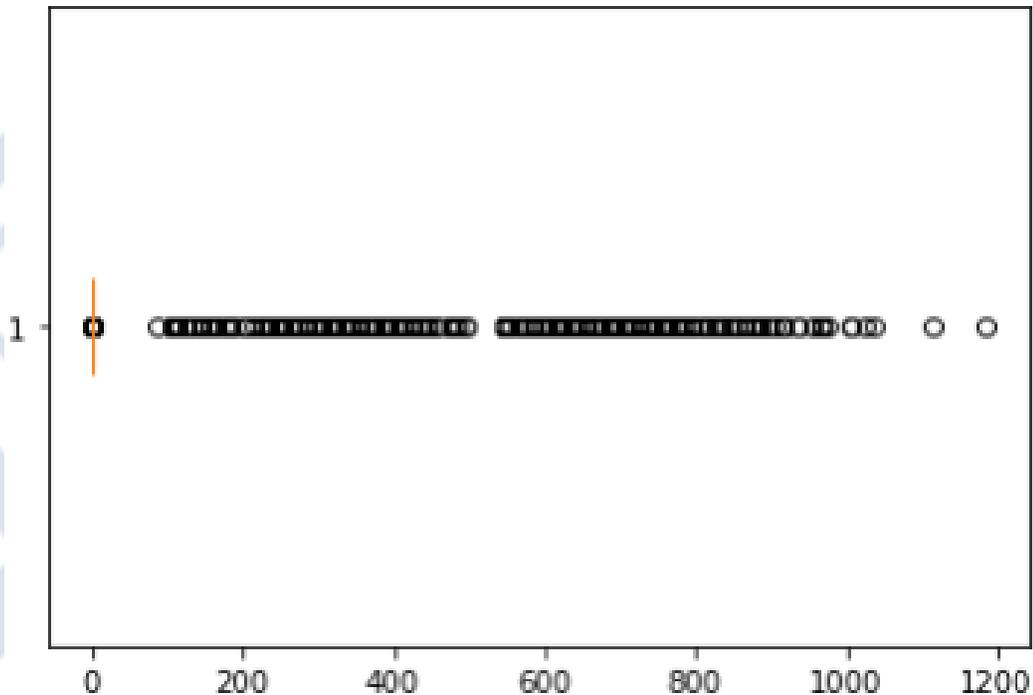


Figura 6. Diagrama de Caja y Bigotes Acidez variable.

## La identificación de los valores outliers se realizó de la siguiente manera:

- Se obtuvo los valores del primer cuartil, tercer cuartil, rango inter cuartil, y mediana, Todo esto para hallar los valores inferior y superior del bigote, y seguido de esto reemplazar los valores outliers, hallados.

```
Q1=dataset['Variable'].quantile(0.25)
Q3=dataset['Variable'].quantile(0.75)
IQR=Q3-Q1
mediana=dataset['Variable'].median()
```

Figura 7. Obtención de valores.



- El siguiente paso, para hallar los valores inferior y superior del bigote

```
B_inf=(Q1 -1.5 * IQR)  
B_sup=(Q3 +1.5 * IQR)
```

Figura 8. Obtención de bigotes.

- Como paso final para el tratamiento de datos, procedemos a reemplazar los valores outliers con la mediana, de estos.

```
dataset['Densidad'] = np.where(ubicacion_outliers, mediana, dataset['Densidad'])
```

Figura 9. Reemplazo outliers por mediana.

## Entrenamiento y Validación

- **Procesamiento de Datos**

```
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt
```

Figura 10. Implementación de librerías.

```
%matplotlib inline  
df=pd.read_csv('dataset_train.csv',engine='python', index_col=0)  
df.head()
```

Figura 11. Importación del dataset.



```
x=df[['Tipo','Acidez fija','Acidez variable','Ácido cítrico','Azucar residual','Cloruros','Dióxido de azufre libre','Nivel de di',  
, 'Densidad','pH', 'Sulfatos','Grado de alcohol' ]]
```

Figura 12. Variables independientes.

```
y=df[('Calidad')]  
y
```

Figura 13. Variable dependiente.

```
model=Sequential()  
model.add(Dense(48, input_dim=12, activation='relu' ))  
model.add(Dense(64, activation='relu'))  
model.add(Dense(10, activation='softmax'))  
model.compile(loss='sparse_categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

```
model.fit(x, y, epochs=10000, batch_size=64)
```

Figura 14. Aplicación de capas.

## Entrenamiento

Para poder entrenar el modelo ya establecido se necesitó de 10000 iteraciones con las cuales se pudo llegar al resultado de una precisión del 81.75%.

```
scores = model.evaluate(x, y)  
print("\n%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))  
153/153 [=====] - 0s 979us/step - loss: 0.4490 - accuracy: 0.8175  
accuracy: 81.75%
```

Figura 15. Entrenamiento.



## Validación

Posteriormente se cargaron los datos de la dataset de prueba para poder evaluar nuestro modelo ya planteado con anterioridad.

```
scores_test = model.evaluate(x_test, y_test)
print("%s: %.2f%%" % (model.metrics_names[1], scores_test[1]*100))
```

51/51 [=====] - 0s 946us/step - loss: 4.8618 - accuracy: 0.5606  
accuracy: 56.06%

Figura 16. Validación.

## CONCLUSIONES

- Usualmente se usan redes neuronales para el tratamiento de datos.
- Las redes neuronales tienen la capacidad de ser tolerantes a fallos y son los primeros métodos computacionales con esta capacidad inherente.
- La evolución de la tecnología con redes neuronales busca asimilarse a un cerebro humano con funciones de aprendizaje y así poder analizar una vasta cantidad de datos mucho más rápido y minimizando errores.
- Los algoritmos para el análisis de datos definen el resultado de un análisis en donde se califican los errores con el fin de reducirlos y hacer más eficiente al sistema.

## REFERENCIAS

- [1] Python.[Online]. Available: <https://es.wikipedia.org/wiki/Python>
- [2] E. Lazarte, "Entrenamiento de una red neuronal hardware desde matlab (hardware in the loop)," Madrid, Febrero, 2017. Available: [http://oa.upm.es/45516/1/TFG\\_EDUARDO\\_LASARTE\\_ZAPATA.pdf/](http://oa.upm.es/45516/1/TFG_EDUARDO_LASARTE_ZAPATA.pdf/).
- [3] D. Peralta, A. Herrera, F. Herrera, "Un estudio sobre el preprocesamiento para Redes Neuronales Profundas y Aplicación sobre Reconocimiento de Dígitos Manuscritos," September 2016. [Online], Available:



<https://eusal.es/index.php/eusal/catalog/download/978-84-9012-632-5/5256/5486-1?inline=1>

- [4] Keras.[Online].Available: <https://keras.io/>
- [5] Tensorflow.[Online]. Available:<https://www.tensorflow.org/>
- [6] Machine Learnig con tensorflow y keras en python.[Online]. Available: <https://www.luisllamas.es/machinelearning-con-tensorflow-y-keras-en-python/>
- [7] Anaconda Distribution: La Suite más completa para la Ciencia de datos con Python.[Online]. Available: <https://blog.desdelinux.net/ciencia-de-datos-con-python/>
- [8] Numpy.[Online].Available: [https://damianavila.github.io/Python-Cientifico-HCC/3\\_NumPy.html](https://damianavila.github.io/Python-Cientifico-HCC/3_NumPy.html)
- [9] Bioinformatics at COMAV.[Online].Available <https://bioinf.comav.upv.es/courses/linux/python/pandas.html>
- [10] Matplotlib. [ Online].Available: <https://es.wikipedia.org/wiki/Matplotlib>.
- [11] Jupyter notebook: documenta y ejecuta código desde el navegador.[Online].Available <https://blog.desdelinux.net/jupyter-notebook/>
- [12] D. Match, "Redes Neuronales: Conceptos Básicos y Aplicaciones," Universidad Tecnológica Nacional – Facultad Regional Rosario Departamento de Ingeniería Química - Grupo de Investigación Aplicada a la Ingeniería Química (GIAIQ), Mar. 2021. [Online] Disponible en: [https://www.fro.utn.edu.ar/repositorio/catedras/quimica/5\\_ano/orientadora1/monografias/match-redesneuronales.pdf](https://www.fro.utn.edu.ar/repositorio/catedras/quimica/5_ano/orientadora1/monografias/match-redesneuronales.pdf)
- [13] J. Montañó, "Redes Neuronales Artificiales aplicadas al Análisis de Datos," UNIVERSITAT DE LES ILLES BALEARS - Facultad de Psicología, 2002. [Online] Disponible en:<https://www.tesisenred.net/bitstream/handle/10803/9441/tjmm1de1.pdf?sequence=1>
- [14] A. Serrano, E. Soria, J. Martín, "Redes neuronales artificiales," Escola Tecnica Superior de Enginyeria - Departament de Enginyeria Electronica, 2010, [Online] Disponible en: [http://ocw.uv.es/ingenieria-y-arquitectura/1-2/libro\\_ocw\\_libro\\_de\\_redes.pdf](http://ocw.uv.es/ingenieria-y-arquitectura/1-2/libro_ocw_libro_de_redes.pdf)



- [15] F. Lara, "Fundamentos de redes neuronales artificiales," Laboratorio de cibernética aplicada, centro de instrumentos UNAM, [Online] Disponible en: [http://conceptos sociales.unam.mx/conceptos\\_final/598trabajo.pdf](http://conceptos sociales.unam.mx/conceptos_final/598trabajo.pdf)
- [16] R Gonzales, "Phyton para todos," [Online] Disponible en: [http://www.utic.edu.py/citil/images/Manuales/Python\\_para\\_todos.pdf](http://www.utic.edu.py/citil/images/Manuales/Python_para_todos.pdf)
- [17] C. Antona, "Herramientas modernas en redes neuronales: La librería Keras," Universidad autónoma de Madrid, Ene. 2017, [Online] Disponible en: [https://repositorio.uam.es/bitstream/handle/10486/677854/antona\\_cortes\\_carlos\\_tfg.pdf?sequence=1&isAll owed=y](https://repositorio.uam.es/bitstream/handle/10486/677854/antona_cortes_carlos_tfg.pdf?sequence=1&isAll owed=y)
- [18] D. Conde, "Inteligencia Artificial con Tensor Flow para la predicción de comportamientos," Departamento de Ingeniería telemática – Escuela Técnica Superior de Ingeniería – Universidad de Sevilla, 2018, [Online] Disponible en: <https://pdfs.semanticscholar.org/bf01/10c9822592997a8ecef35bb3963e86276c54.pdf>
- [19] Rip Tutorial, "Aprendizaje pandas," [Online] Disponible en: <https://riptutorial.com/Download/pandas-es.pdf>
- [20] F. Batista, "Numpy + Scipy", [Online] Disponible en: <http://www.taniquetil.com.ar/homedevel/presents/numsci.pdf>
- [21] J. Padilla, "Gráficas con la librería Matplotlib para Python," [Online] Disponible en: [http://jpadilla.docentes.upbbga.edu.co/Logica\\_y\\_Algoritmia/Graficas%20con%20la%20Olibreria%20Matplot lib%20para%20Python.pd](http://jpadilla.docentes.upbbga.edu.co/Logica_y_Algoritmia/Graficas%20con%20la%20Olibreria%20Matplot lib%20para%20Python.pd)

# Revisión de modelos que integren Design Thinking en metodologías de Desarrollo Ágil

*Review of models that integrate Design Thinking in Agile Development methodologies*

47



**Dewitt Scott Chavez Ponce**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.  
@ [dchavezp@unsa.edu.pe](mailto:dchavezp@unsa.edu.pe)

**Robert Teodoro Arce Apaza**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.  
@ [rarcea@unsa.edu.pe](mailto:rarcea@unsa.edu.pe)

**Andrea Flores Choquehuanca**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.  
@ [aflorescho@unsa.edu.pe](mailto:aflorescho@unsa.edu.pe)

**Daniel Augusto Prado Cussi**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.  
@ [dpradocu@unsa.edu.pe](mailto:dpradocu@unsa.edu.pe)

**Mario Huaypuna Cjuno**  
Universidad Nacional de San Agustín.  
Arequipa, Perú.  
@ [mhuaypuna@unsa.edu.pe](mailto:mhuaypuna@unsa.edu.pe)

 **ARK:** [ark:/42411/s8/a52](https://nbn-resolving.org/urn:nbn:pe:unsa-1-42411-s8-a52)

 **PURL:** [42411/s8/a52](https://nbn-resolving.org/urn:nbn:pe:unsa-1-42411-s8-a52)

RECIBIDO 30/10/2021 • ACEPTADO 03/12/2021 • PUBLICADO 30/03/2022

## RESUMEN

El presente artículo tiene como finalidad de brindar un repaso de la integración de Design Thinking en el desarrollo de software, usando la metodología ágil de trabajo SCRUM. Se intenta añadir técnicas Scrum a una clase de Design Thinking basada en proyectos. El objetivo principal es brindar técnicas de administración de proyectos útiles y flexibles que encajen bien con el resto del kit de herramientas Design Thinking. Como resultados se observa que Scrum durante el inicio de los proyectos junto con mucha otra información nueva, no daba suficiente tiempo a los participantes para asimilarlo todo por completo, a pesar de ello los estudiantes parecieron gustar del uso de scrum y sus técnicas.

**Palabras claves:** Design Thinking, Metodologías ágiles, SCRUM.

## ABSTRACT



*The article exposed aims to make a review of Design Thinking (DT) Integration in software development, using the SCRUM methodology. It tries adding Scrum techniques to DT class based on projects. The main objective is to provide useful project management that fit well with the rest of the DT tools kit. As a result, it is observed that Scrum during the start of the projects, together with a lot of other new information, did not give participants enough time to assimilate everything completely, although the students seemed to like the use of scrum and its techniques.*

**Keywords:** Agile Methodologies, Design Thinking, SCRUM.

---

## INTRODUCCIÓN

Design Thinking se ha vuelto popular como una metodología que produce soluciones innovadoras y creativas en diferentes industrias, por ejemplo, en la industria del software o en el desarrollo de productos. Si bien Design Thinking ya proporciona un proceso junto con varias técnicas para usar en cada paso del proceso, carece de técnicas de gestión de proyectos. [7]

Scrum es la metodología de desarrollo de software ágil más popular y recientemente también ha llegado a otras industrias. Proporciona un marco que permite a un equipo planificar e implementar su trabajo en ciclos más pequeños, llamados sprints, que duran entre 1 y 4 semanas. Las técnicas de planificación y reflexión utilizadas en los proyectos Scrum son fáciles de entender e implementar y encajan con la mentalidad general de Design Thinking.

El presente artículo tiene la finalidad de brindar un repaso de la integración de Design Thinking en el desarrollo de software, principalmente en lo que respecta al uso de marcos de trabajo para el proceso de desarrollo y más específicamente el marco de trabajo SCRUM.

## Conceptos Fundamentales

Para la presente investigación se revisaron artículos en buscadores y bases de datos científicas como Scholar de Google, Web of Science, Scopus, IEEE, Science Direct. En las siguientes secciones se presenta conceptos clave como SCRUM y Design Thinking (DT), luego se detalla la forma en cómo distintos autores han integrado Design Thinking en SCRUM. Debido a que la presente investigación tiene como objetivo ver el panorama sobre la integración de SCRUM con DT se van a presentar tablas comparativas hechas en que indicarán a manera de resumen la forma en la que han integrado SCRUM y DT.



## Design Thinking

Método para generar ideas innovadoras centradas en la eficacia de entender y dar solución a las necesidades reales de los usuarios.

El proceso consiste en las siguientes etapas:

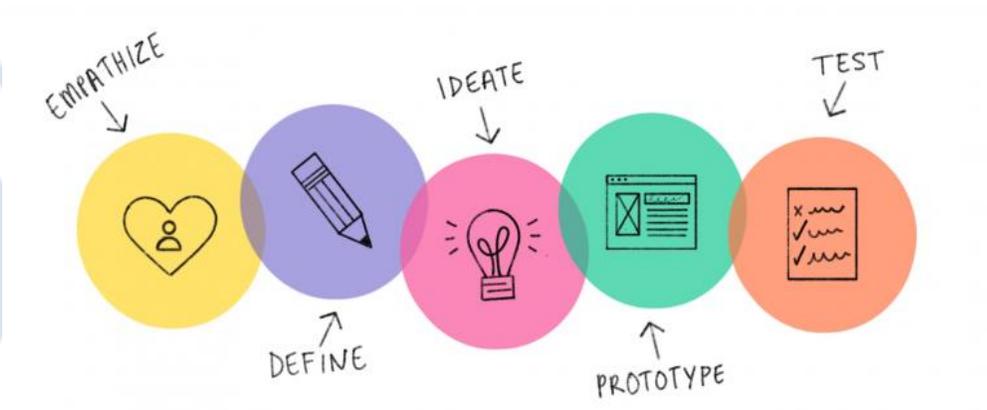


Figura 1 Procesos de Design Thinking Fuente: designthinking.es

- **EMPATIZA:** Abarca la profunda comprensión de las necesidades de los usuarios para generar soluciones reales.
- **DEFINE:** Escoge la información recopilada que aporta valor y lleva al alcance de nuevas perspectivas. Se identifican problemas cuyas soluciones serán clave para la obtención de un resultado innovador.
- **IDEA:** Tiene como objetivo la generación de opciones.
- **PROTOTIPA:** Se construyen prototipos que ayudan a visualizar las posibles soluciones, poniendo de manifiesto elementos que debemos mejorar o refinar antes de llegar al resultado final.
- **TESTEA:** Se prueban los prototipos con los usuarios implicados. Esta fase es crucial ya que ayuda identificar mejoras significativas, fallos a resolver, posibles carencias.



## Metodología Ágil SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas centradas en el trabajo colaborativo, en equipo, para obtener el mejor resultado posible de un proyecto.

El proceso consiste en lo siguiente:

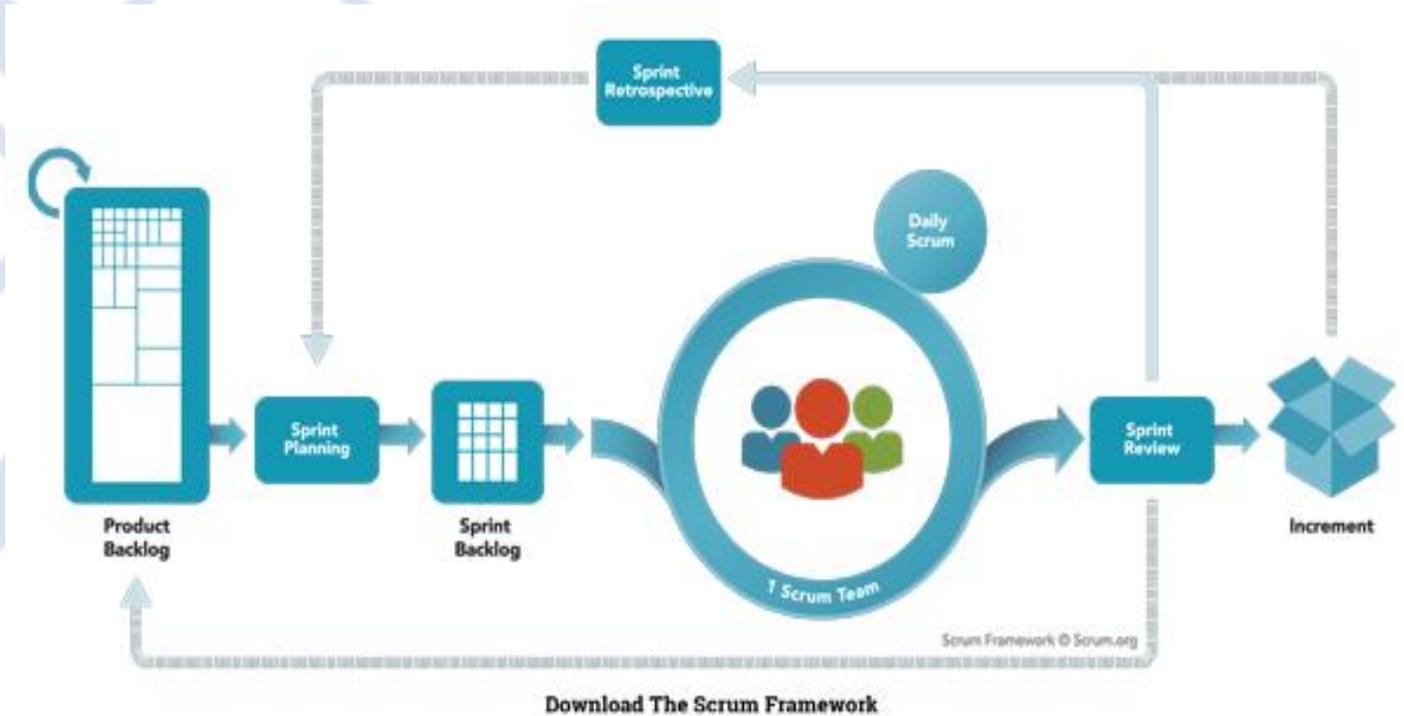


Figura 2 Procesos SCRUM Fuente: Scrum.org

El proceso se puede definir como un periodo de tiempo de un mes o menos en el que se crea un producto liberable, utilizable y "Finalizado".

Los sprints deberían tener duraciones constantes durante todo el desarrollo. Un nuevo Sprint comienza solo cuando el anterior ha finalizado. Dentro de los agentes tenemos a: Product Owner, Scrum Master y el Development team.



## Integración de Design Thinking con SCRUM

### Sinergia Entre Metodologías

Ambas metodologías poseen herramientas que puede complementarse una con la otra [1], como se ve en la figura siguiente, Design Thinking está enfocado al usuario, por lo tanto se adecua perfectamente a las primeras fases de un proyecto de software, que es recopilar las necesidades del usuario y convertirlas en requisitos del proyecto, para ello son útiles técnicas como: la Observación Encubierta y Focus Group [8].

Cabe resaltar la relación de las etapas IDEAR y PROTOTIPAR de Design Thinking, con las actividades de definición de requisitos y prototipado en un proceso de desarrollo de software orientado a prototipos. Dichos prototipos deberían recibir el visto bueno del usuario, para que luego dichos prototipos pasen a la fase de implementación donde entra en juego SCRUM [10].

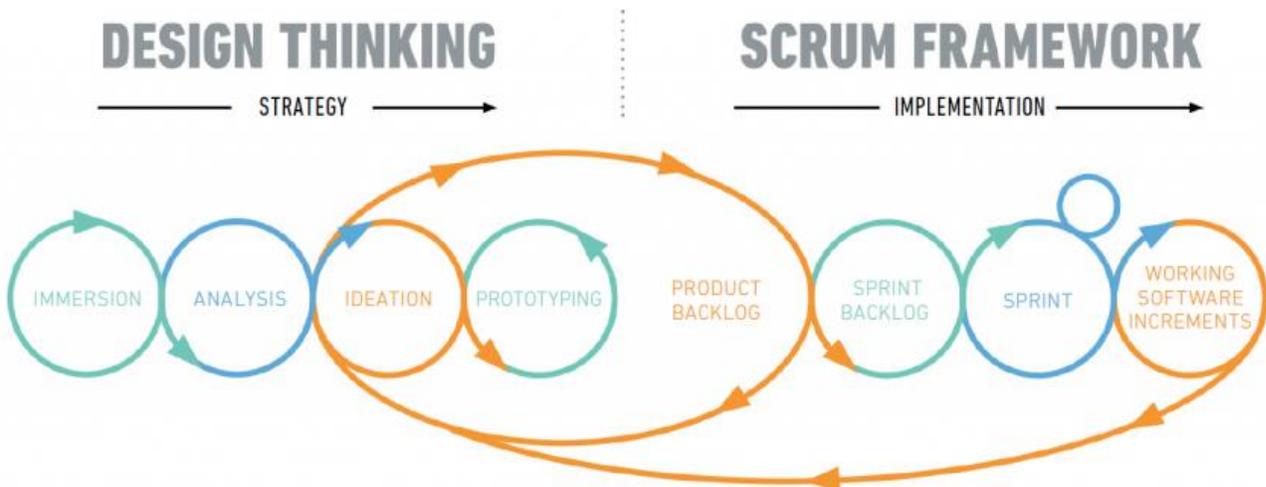


Figura 3. Sinergia entre Design Thinking y SCRUM. Fuente [10]

## Modelo Basado en DT y Metodologías Ágiles para Proyectos de Juegos Digitales

Los autores indican que el enfoque de DT se puede aplicar al aspecto creativo del proceso de desarrollo del juego, mientras que las metodologías ágiles se pueden utilizar en la creación de prototipos y el desarrollo del producto final. A continuación, se muestra un cuadro resumen de la sinergia entre DT y Metodologías ágiles



En esta propuesta se centra en usar DT en los procesos iniciales de concepción y preproducción y las metodologías ágiles para los procesos de producción, post producción y post ejecución.

Para evaluar su propuesta hicieron entrevistas a desarrolladores seleccionados por teniendo en cuenta dos criterios: experiencia en el desarrollo del juego y el papel principal que desarrolla en la empresa. A partir de ello concluyeron lo siguiente [3]:

- El desarrollo se adapta a las características DT como: empatía, pensamiento integrador, optimismo, experimentación y colaboración.
- Los desarrolladores no tenían conocimiento de nomenclaturas de DT.
- Las técnicas DT han sido utilizados por los desarrolladores principalmente con aspectos creativos del desarrollo del juego, y como se esperaba, que ellos se utilizan sobre todo en las etapas de concepción y preproducción.
- El modelo está enfocado en dos áreas: la primera en la que interviene DT y la segunda en la que intervienen las metodologías ágiles. La idea está centrada en que DT sirva de apoyo en determinar la visión del producto y el product backlog que será usada posteriormente en SCRUM.

## **DT@SCRUM**

Los autores plantean usar Scrum que proporciona el marco general para el desarrollo y las actividades del DT es decir los equipos utilizarán Sprint para estructurar sus actividades durante el desarrollo de software y las actividades de diseño.

Con el fin de permitir que los equipos de diseño hacerse una idea de la duración y el valor del diseño actividades DT para que puedan estructurar mejor su trabajo creativo.

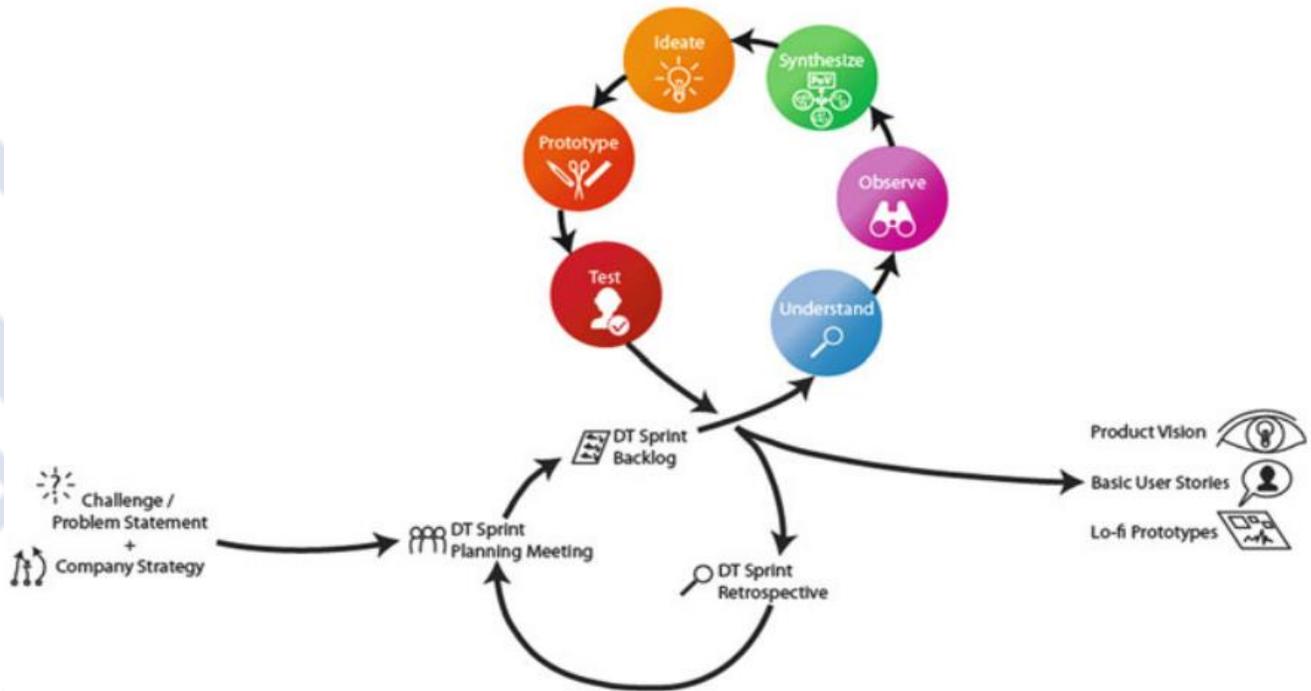


Figura 3 Design Thinking Mode Fuente: D. Sirkin

El equipo comienza con la fase entender y observar para que toda la información recopilada durante las dos primeras fases se sintetice en el punto de vista del equipo. Sobre esta base el equipo de diseño idea aspectos de una posible solución. Luego las ideas generadas serán prototipadas que luego se someterán a las pruebas con usuarios de destino.

La información obtenida mediante pruebas de las ideas tiene que ser sintetizada de nuevo. Dependiendo del resultado el equipo iniciará una iteración consecutivo en el que se va a seguir adelante con más ideas para refinar la idea o, volver a entender y observar las fases para responder a preguntas abiertas e investigar nuevos aspectos del problema.

Los autores resaltan el concepto de Planificación de Diseño para la ejecución de tareas de diseño en Sprint cortos, estimar y planificar en consecuencia, y usando una retrospectiva regular para ayudar a comprender mejor el proceso y las tareas de diseño.

Concluyen que DT @ Scrum es un enfoque que conecta a la perfección la generación de ideas innovadoras y su implementación [4].



## IBM DT

Se extiende el método DT original, proporcionando un nuevo enfoque para escribir los requisitos, la organización de los equipos, y realizar un seguimiento del progreso del proyecto incluyendo la retroalimentación del usuario final durante todas las fases de desarrollo del proyecto.

Se definen tres roles principales: El Product Manager, Sponsor User y el Diseñador que tienen el control de la arquitectura del proyecto y código ejecutable, prototipo y el plan de carrera técnica.

El IBM DT involucra dos fases: La fase de visión para formular requisitos software a través del uso de varias prácticas de DT que combina las user personas, mapas de empatía, Hills y mapas historia. La Delivery Wave consiste en sprints de desarrollo de software conducidos por equipos multidisciplinarios que incluye Sponsor Users que contribuyen con información constante acerca de los artefactos entregados.

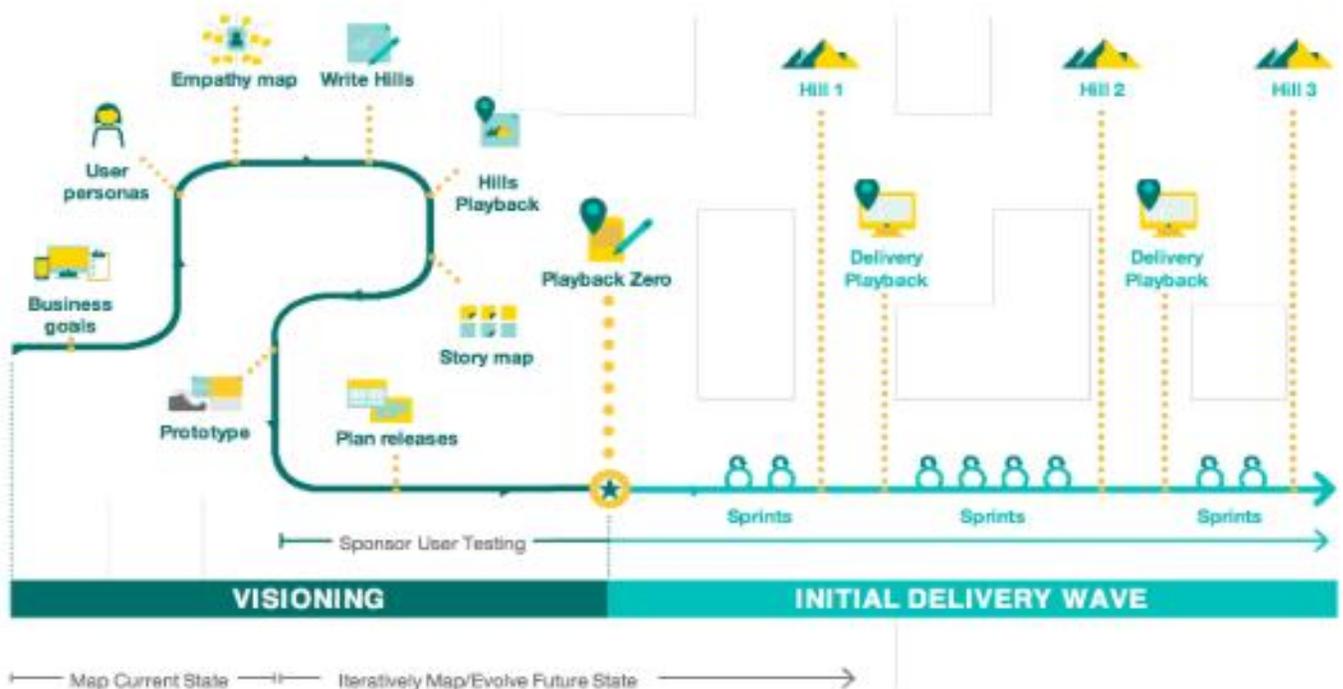


Figura 4 IBM DT Framework Fuente: Lucena

Los componentes adicionales respecto al DT son los siguientes [5]:



Los Hills son forma de expresar necesidades de los usuarios en los requisitos del proyecto para cumplir con un problema de usuario específico, claramente definido que está informado por la investigación de usuarios.

Los Sponsor Users que son personajes pueden representar sólo una parte de la comprensión de las necesidades del usuario

Playbacks que son puestos de control cuando el equipo del proyecto y el Sponsor User se reúnen con el fin de examinar el estado del proyecto y planificar los pasos siguientes.

## **Agregar gestión de proyectos al estilo Scrum a Design Thinking [7]**

En este artículo se explora la adición de técnicas Scrum a una clase de Design Thinking basada en proyectos. El objetivo principal es brindar técnicas de administración de proyectos útiles y flexibles que encajen bien con el resto del kit de herramientas Design Thinking. Una de las principales dificultades encontradas en esta aplicación fue que el hecho de presentar Scrum durante el inicio de los proyectos junto con mucha otra información nueva, no daba suficiente tiempo a los participantes para asimilarlo todo por completo; a pesar de ello los estudiantes parecieron gustar del uso de scrum y sus técnicas.

## **Design Thinking integrado en el desarrollo de software [9]**

Este estudio tiene como objetivo evaluar cómo se integran el enfoque DT junto con el Desarrollo de software ágil. Para este propósito el autor a optado por evaluar múltiples modelos de aplicación de DT, obteniendo resultados que reflejan la gran variedad de estas aplicaciones.

En general el uso de un enfoque de DT promueve la comunicación entre los equipos y los clientes a lo largo de todo el proyecto de desarrollo de software. Además, se observa que en algunos casos la calidad del software aumentó significativamente y que se percibió la satisfacción de los usuarios o clientes del software. Los aspectos del enfoque de DT se caracterizan por la empatía, la definición y la creación rápida de prototipos. Esto es crucial para un estrechamiento entre el equipo y los destinatarios del resultado del proyecto, verificando los factores técnicos y no técnicos.

## **Comparativa**

A continuación, se presenta una tabla comparativa de las similitudes y diferencias de los modelos en los que se integra DT en metodologías ágiles.



Tabla 1. Diferencias y similitudes entre modelos que integran DT y metodologías ágiles

Diferencias	Similitudes
<ul style="list-style-type: none"><li>- La metodología DT para juegos es concebida de manera por dos áreas por separado pero que se intersecan para definir la visión y el product backlog en comparación con DT@ SCRUM e IBM DT que las actividades se dan en cada Sprint</li></ul>	<ul style="list-style-type: none"><li>- DT@SCRUM e IBM DT las actividades DT se dan en Sprints</li></ul>
<ul style="list-style-type: none"><li>- IBM DT integra nuevos roles y actividades respecto al modelo DT@SCRUM y la metodología DT para juegos que usan las actividades base de DT</li></ul>	

## DISCUSIÓN Y CONCLUSIONES

Tanto el Scrum como el Design Thinking requieren de grupos proactivos en el proyecto donde se desempeñen. Entre otras similitudes, por un lado, vemos que la iteración (Sprint) en Scrum es un proceso reiterativo en el que cada entregable terminado es testeado y retroalimentado por la participación del Product Owner.

Por otro lado, entre las etapas de prototipado y testeo del Design Thinking hay un ciclo Iterativo en el que la idea de producto se enriquece mediante el aporte del cliente y la capacidad creativa del grupo de trabajo.

Por lo tanto, el Design Thinking puede aprovechar los roles e instrumentos del Scrum sin dejar de usar las herramientas que permiten conocer al cliente a profundidad. Esto permitirá el desarrollo de proyectos en períodos más reducidos que toman en cuenta lo que el consumidor demanda. Por otro lado, los equipos que utilizan Scrum se benefician de las herramientas que se utilizan en el Design Thinking (sobre todo en las etapas de definición y prototipado) para conocer con mayor profundidad las necesidades del cliente sin dejar el enfoque de gestión ágil que permite la ejecución del proyecto. Se ha visto que la integración de DT en la metodología SCRUM ayuda a entender de mejor manera las necesidades del cliente ya sea que se tome en la etapa inicial de SCRUM como lo hizo Higuchi en su metodología DT para juegos o en cada SPRINT como lo desarrollan IBM DT y DT@SCRUM.

## REFERENCIAS

- [1] E. C. Silva, "Metodologías ágiles y design thinking: Gestión efectiva basada en las necesidades e intereses de los clientes" InnoVaG, no. 1, pp. 14-16, 2016.



- [2] G. Pizarro, "Estudio descriptivo de las prácticas de diseño y arquitectura de desarrollo de software en las compañías startups." *Revista Ciencia UNEMI*, vol. 10, no. 23, pp. 125–132, 2017.
- [3] M. M. Higuchi and D. N. Nakano, "Agile Design: A Combined Model Based on Design Thinking and Agile Methodologies for Digital Games Projects," *Rev. Gestão e Proj.*, vol. 08, no. 02, pp. 109–126, 2017.
- [4] F. Häger, T. Kowark, J. Krüger, C. Vetterli, F. Übernickel, and M. Uflacker, "DT@Scrum: Integrating Design Thinking with Software Development Processes," Springer, Cham, 2015, pp. 263–289.
- [5] P. Lucena, A. Braz, A. Chicoria, and L. Tizzei, "IBM design thinking software development framework," *Commun. Comput. Inf. Sci.*, vol. 680, pp. 98–109, 2017.
- [6] DOBRIGKEIT, Franziska, et al. Adding Scrum-style project management to an advanced Design Thinking class. DS 91: Proceedings of NordDesign 2018, Linköping, Sweden, 14th–17th August 2018, 2018.
- [7] ULRICH FRYE, Maria Elmquist, The Integration of Design Thinking and Lean Software Development from the Perspective of Product Owners and Scrum Masters: Proceedings Göteborg, Sweden 2013
- [8] J. S. Apatet al., "Aplicaciones móviles para estudiantes a través de design thinking y scrum," 2017.
- [9] PEREIRA, Julio Cesar; DE FSM RUSSO, Rosaria. Design Thinking Integrated in Agile Software Development: A Systematic Literature Review. *Procedia computer science*, 2018, vol. 138, p. 775-782.
- [10] "Optimizing projects with design thinking and scrum" Apr 2018.[Online]. Available: <http://blog.mjvinnovation.com/design-thinking-2/design-thinking-and-scrum-associated-methodologies-drive-business-and-deliver-valuable-projects>
- [11] R. Goodspeed, C. Riseng, K. Wehrly, W. Yin, L. Mason, and B. Schoenfeldt, "Applying design thinking methods to ecosystem management tools: Creating the great lakes aquatic habitat explorer," *Marine Policy*, vol. 69, pp. 134–145, 2016.



# Uso de árboles de decisión para detectar si una habitación está ocupada usando Python

58

*Use of decision trees to detect if a room is occupied using Python*

## **Joel Atamari Aguilar**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [jatamaria@unsa.edu.pe](mailto:jatamaria@unsa.edu.pe)

## **Christian Flores Conde**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [cfloresc@unsa.edu.pe](mailto:cfloresc@unsa.edu.pe)

## **Jhon Mamani Mamani**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [jmamanim@unsa.edu.pe](mailto:jmamanim@unsa.edu.pe)

## **Sergio Rondon Polanco**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [srondonp@unsa.edu.pe](mailto:srondonp@unsa.edu.pe)

 **ARK:** [ark:/42411/s8/a53](https://nbn-resolving.org/ark:/42411/s8/a53)

 **PURL:** [42411/s8/a53](https://nbn-resolving.org/ark:/42411/s8/a53)

RECIBIDO 18/11/2021 • ACEPTADO 22/12/2021 • PUBLICADO 30/03/2022

## **RESUMEN**

En este artículo se presenta una descripción de los árboles de decisión para determinar si una habitación está ocupada o no. En esta investigación se demuestra empíricamente que es posible determinar si una habitación está ocupada o no, usando las variables temperatura, humedad, luminosidad, nivel de CO<sub>2</sub> y el radio de humedad, mediante la utilización de árboles de decisión con las librerías SKLEARN en el lenguaje Python.

**Palabras claves:** Inteligencia artificial, árboles de decisión, CO<sub>2</sub>, etiquetas, Python.

## **ABSTRACT**

*This article presents a description of the decision trees for determining whether a room is occupied or not. In this research it is empirically demonstrated that it is possible to determine whether a room is occupied or not, using the variables temperature, humidity, luminosity, CO<sub>2</sub> level and the humidity ratio, by using decision trees with the SKLEARN libraries in the language Python.*

**Keywords:** Artificial Intelligence, Decision trees, CO<sub>2</sub>, labels, Python.



## INTRODUCCIÓN

Según Marvin Minsky [1], la inteligencia artificial (IA) es la ciencia de construir máquinas para que hagan cosas que, si las hicieran humanos, requerirían inteligencia.

La IA tiene diferentes técnicas como sistemas expertos basados en reglas, redes neuronales artificiales, árboles de decisión, etc. [2]. A continuación, se presenta la implementación de un árbol de decisión para determinar si una habitación está ocupada o no con un dataset de más de 20000 registros utilizando una biblioteca de software automático llamada Sklearn.

## Árboles de decisión

Los árboles de decisión es una de las técnicas de aprendizaje inductivo supervisado no paramétrico, se utiliza para la predicción y se emplea en el campo de inteligencia artificial, donde a partir de una base de datos se construyen diagramas de construcción lógica, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren en forma repetitiva para la solución de un problema.[3]

Nuestra inclinación por el uso de esta técnica se debe a sus ventajas:

- Se puede trabajar con valores cuantitativos (solo tuvimos que transformarlos).
- Al contar un número pequeño de características (cinco) es factible el uso de esta técnica.

## Limpieza de datos

Inicialmente nuestra base de conocimiento estaba conformada por 7 columnas (No, Fecha, Temperatura, Humedad, Luminosidad, Nivel de CO2, Radio de humedad y la salida). Además, esta contaba con 20560 casos.

Con el objetivo de realizar un mejor entrenamiento se realizó la limpieza de datos utilizando como herramienta principal Microsoft Excel 2016. Se eliminó la columna Fecha ya que esta no tiene algún valor significativo para el entrenamiento al ser diferente para cada caso. Se eliminó filas repetidas usando la herramienta antes ya mencionada, con lo que el número de filas se redujo a 19119. Finalmente, no se encontraron valores muy alejados al promedio.



Después de haber realizado la fase de limpieza de datos se continuó con la transformación de estos.

En esta parte necesitamos hacer un mapeo de los datos en donde transformamos estos datos de entrada en valores categóricos. Esto debido a que nuestros datos son cuantitativos y no cualitativos.

Se utilizó la Ecuación 1 para esta transformación, la cual resta el menor valor al mayor valor y lo divide entre la cantidad de categorías. Esto para posteriormente ir sumando el resultado al valor mínimo y definir los rangos de cada categoría.

$$\text{Rango de cada categoría} = (\text{Valor máximo} - \text{Valor mínimo}) / \text{Cantidad de categorías} \quad (1)$$

- **Temperatura** : Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 20.0816666, en la segunda los datos tienen una lectura mayor a 20.0816666 y menor igual a 21.1633332, en la tercera categoría los datos tienen una lectura mayor a 21.1633332 y menor igual a 22.2449998, en la cuarta categoría los datos tienen una lectura mayor a 22.2449998 y menor igual a 23.3266664, finalizando en la quinta categoría donde los datos tienen una lectura mayor a 23.3266664.

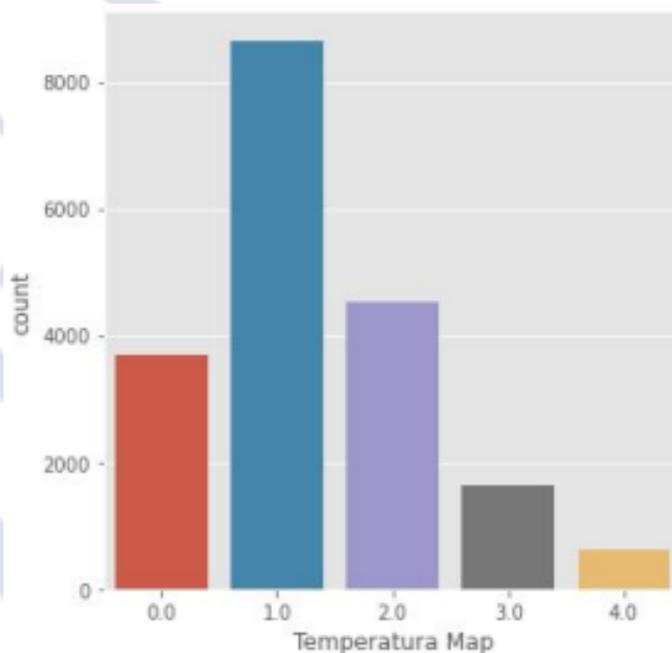


Figura 1. Cantidad de datos de Temperatura dividido en 5 categorías.



- **Humedad:** Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 21.296, en la segunda los datos tienen una lectura mayor a 21.296 y menor igual a 25.847, en la tercera categoría los datos tienen una lectura mayor a 25.847 y menor igual a 30.398, en la cuarta categoría los datos tienen una lectura mayor a 30.398 y menor igual a 34.949, finalizando en la quinta categoría donde los datos tienen una lectura mayor a 34.949.

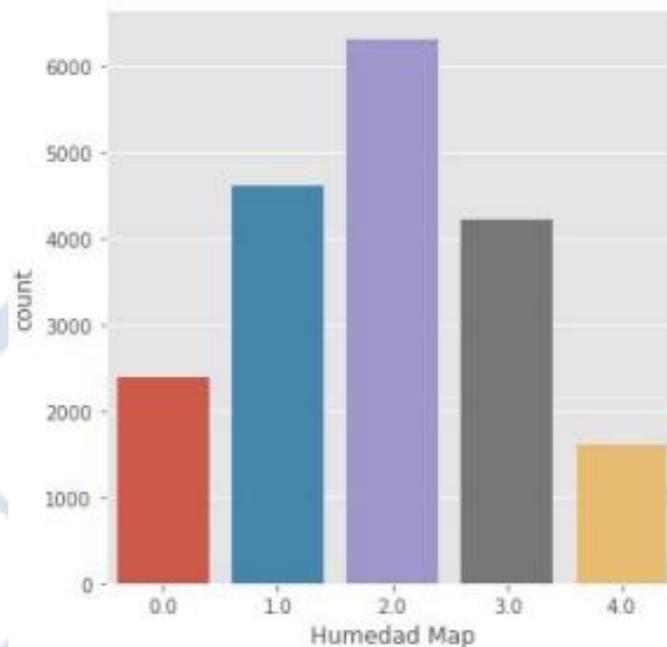


Figura 2. Cantidad de datos de Humedad dividido en 5 categorías.

- **Luminosidad:** Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 339.45, en la segunda los datos tienen una lectura mayor a 339.45 y menor igual a 678.9, en la tercera categoría los datos tienen una lectura mayor a 678.9 y menor igual a 1018.35, en la cuarta categoría los datos tienen una lectura mayor a 1018.35 y menor igual a 1357.8, finalizando en la quinta categoría donde los datos tienen una lectura mayor a 1357.8.

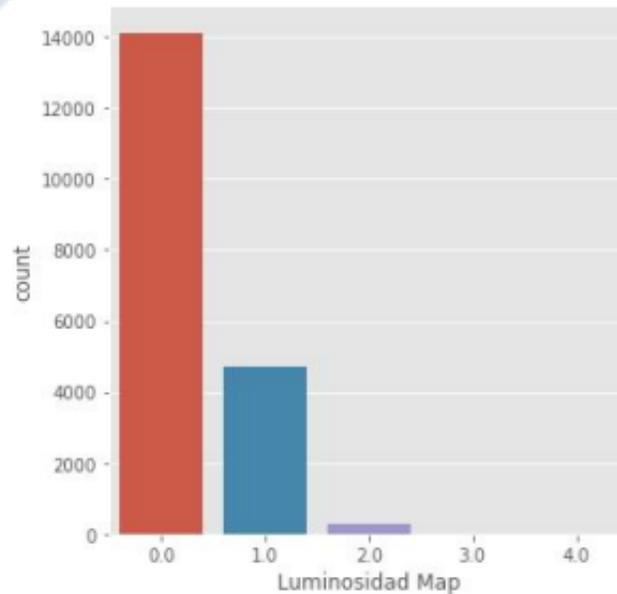


Figura 3. Cantidad de datos de Luminosidad dividido en 5 categorías.

- **Nivel de CO2:** Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 745.5, en la segunda los datos tienen una lectura mayor a 745.5 y menor igual a 1078.25, en la tercera categoría los datos tienen una lectura mayor a 1078.25 y menor igual a 1411, en la cuarta categoría los datos tienen una lectura mayor a 1411 y menor igual a 1743.75, finalizando en la quinta categoría donde datos tienen una lectura mayor a 1743.75.

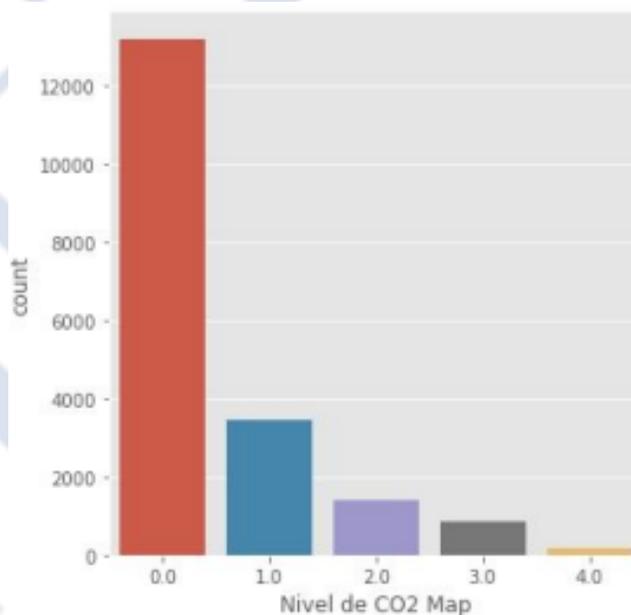


Figura 4. Cantidad de datos de CO2 dividido en 5 categorías



- **Radio de Humedad** :Donde se dividió en 5 categorías, la primera categoría donde los datos tienen una lectura menor igual a 0.0034344, en la segunda los datos tienen una lectura mayor a 0.0034344 y menor igual a 0.0041948, en la tercera categoría los datos tienen una lectura mayor a 0.0041948 y menor igual a 0.0049552, en la cuarta categoría los datos tienen una lectura mayor a 0.0049552 y menor igual a 0.0057156, finalizando en la quinta categoría donde datos tienen una lectura mayor a 0.0057156.

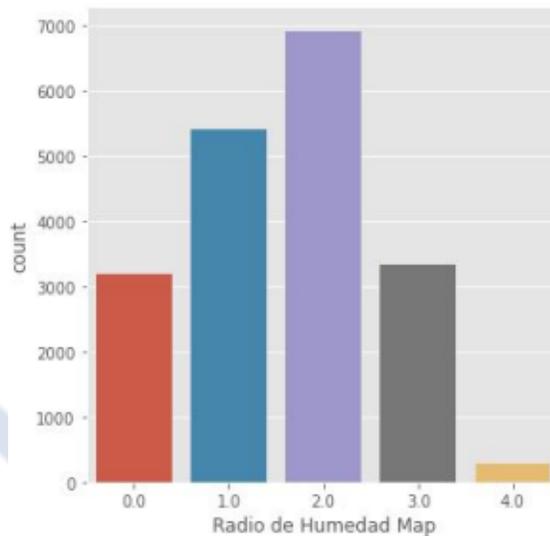


Figura 5. Cantidad de datos de Radio de Humedad dividido en 5 categorías

- **Habitación Ocupada**: Habiendo en estas 2 categorías ,1 para los datos que tienen el valor "si" y 0 para los que tienen el valor "no"

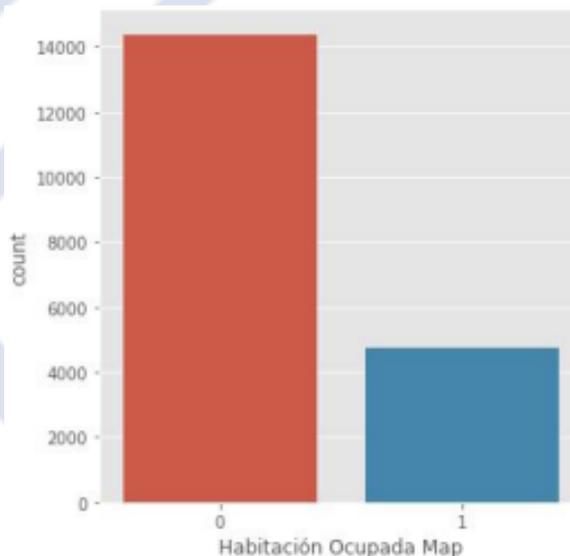


Figura 6. Cantidad de datos de Habitación Ocupada dividido en 2 categorías



Una vez categorizados todos los datos empezamos con la fase de entrenamiento.

## Fase de entrenamiento

Después de haber realizado el mapeo de los datos, se realizó la fase de entrenamiento en el lenguaje Python utilizando Jupyter. Las librerías utilizadas son Numpy, Pandas y Sklearn.

Para la creación del árbol de decisión se utilizaron los siguientes parámetros `criterion="gini"`, `min_samples_split=100`, `min_samples_leaf=20` y `class weight={1:3:22}`.

Respecto a los datos usados para el entrenamiento, se usó el 75% para este y el 25% restante para la fase de comprobación. Se desarrolló la siguiente interfaz para el ingreso de datos:

```
Sensor para detectar si una habitación está ocupada
-----
Ingrese el nombre del dataset (ej. dataset.xlsx): dataset.xlsx
¿Desea cargar un fichero? (s/n): n
-----
Se procederá a realizar el entrenamiento
Porcentaje utilizado para test, el otro porcentaje será utilizado para el entrenamiento (ej. 0.25): 0.25
Se empezó a realizar el entrenamiento ...
Se realizó el entrenamiento y testeo con éxito
-----
```

Figura 7. Interfaz para entrenamiento

## Fase de comprobación

Para la comprobación se trabajó con el 25% de información restante. En esta fase se comprobó la salida esperada con la salida obtenida con lo que cada vez que no coincidan el error acumulado aumenta. Se muestra a continuación la interfaz para la fase comprobación.

```
-----
Resultados
Numero de casos de prueba: 5140
Numero de errores: 55
Porcentaje de error: 1.07 %
¿Desea guardar un fichero con el modelo entrenado? (s/n): s
-----
Se procederá a guardar el fichero
Ingrese el nombre del fichero (sin extensión): train
Se guardó el fichero con el nombre train
¿Desea salir (s/n): s
Vuelva pronto
-----
```

Figura 8. Interfaz para comprobación



## Análisis de resultados

Como se observa en la Figura 8 el porcentaje de error después de la fase de entrenamiento es de 1.07%. A continuación, se muestra una comparación entre los datos esperados y los obtenidos gráficamente. La figura 9 muestra una comparación solo entre los 160 primeros casos para una mejor visualización. En una comparación de los 19119 casos se repetirá.

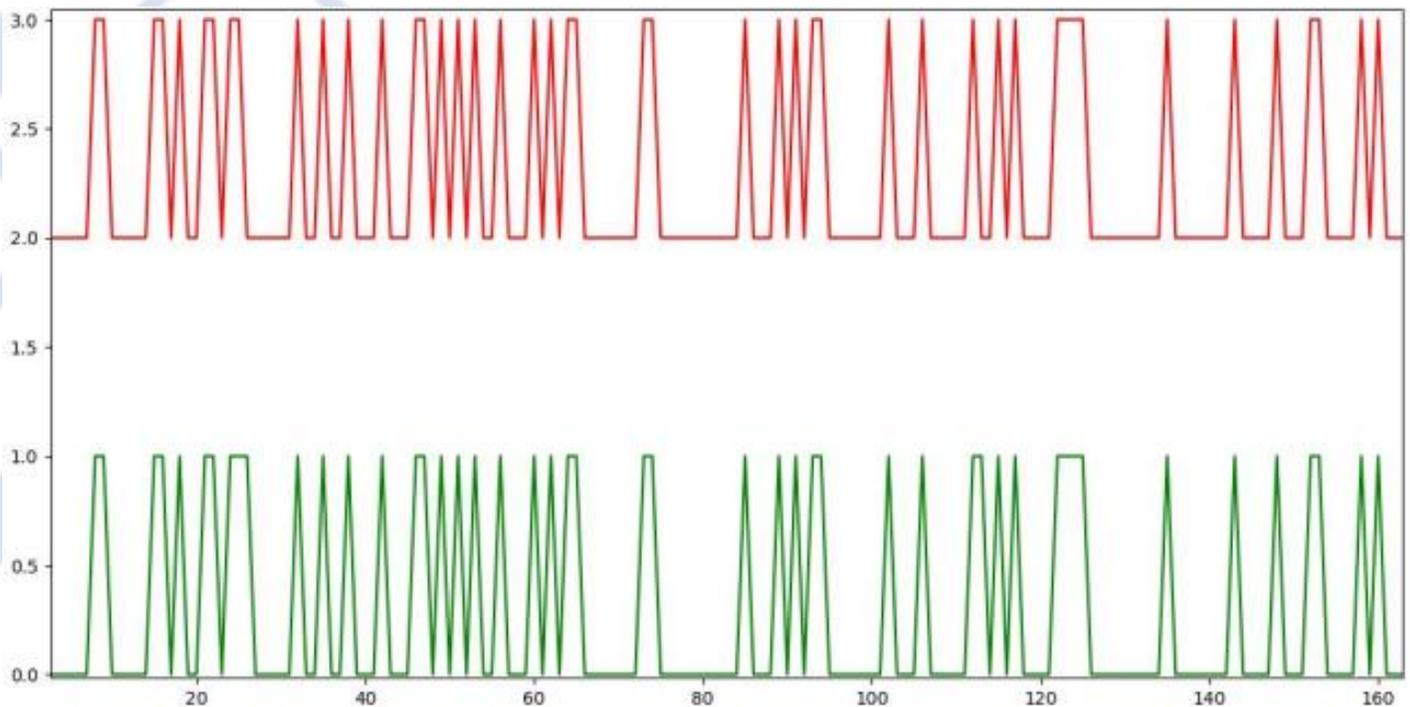


Figura 9. Gráfico comparativo de los resultados esperados (verde) y los datos obtenidos (rojo).

Como trabajo próximo se sugiere la comparación de diversas técnicas de aprendizaje para el mismo contexto.

## CONCLUSIONES

- El uso de árboles de decisión como técnica de entrenamiento para determinar si una habitación está ocupada tuvo 1% de error. Por lo que al usar este algoritmo obtuvimos buenos resultados.



- Los árboles de decisión son una técnica muy fácil de entender y aplicar a diversos contextos, ya que, para este caso, aunque no tuviésemos valores cualitativos pudimos utilizarla sin problemas gracias a una adecuada transformación de datos.
- El uso de Python y sus librerías facilitan la aplicación de muchas técnicas de inteligencia artificial.

## REFERENCIAS

- [1] M. Minsky, «The age of Intelligent Machines: Thoughts About Artificial Intelligence,» KurzweilAI.net., 1990.
- [2] J. De Andres Oviedo, «Técnicas de inteligencia artificial aplicadas al análisis de la solvencia empresarial,» Universidad de Oviedo, Oviedo, 2000.
- [3] G. R. Solarte Martínez y J. Soto Mejía, «Árboles de decisiones en el diagnóstico de enfermedades cardiovasculares,» Scientia et Technica, vol. XVI, nº 49, pp. 104-109, 2011.



# Design Thinking para resolver problemas con la selección de métricas relacionadas a la Calidad del Software

67

*Design Thinking to Solve Problems with the Selection of Metrics in Software Quality*

**Eduardo A. Sutta Gonzales**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [esuttaq@unsa.edu.pe](mailto:esuttaq@unsa.edu.pe)

**Christian L. Salazar Bobadilla**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [csalazarb@unsa.edu.pe](mailto:csalazarb@unsa.edu.pe)

**Julio Cesar Aguilar Uñapilco**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [jaquilaru@unsa.edu.pe](mailto:jaquilaru@unsa.edu.pe)

**David Huittocollo Quispe**

Universidad Nacional de San Agustín.  
Arequipa, Perú.

@ [dhuittocollo@unsa.edu.pe](mailto:dhuittocollo@unsa.edu.pe)

 **ARK:** [ark:/42411/s8/a54](https://nbn-resolving.org/ark:/42411/s8/a54)

 **PURL:** [42411/s8/a54](https://nbn-resolving.org/42411/s8/a54)

RECIBIDO 30/11/2021 • ACEPTADO 05/01/2022 • PUBLICADO 30/03/2022

## RESUMEN

En el presente artículo se va a utilizar la técnica Design Thinking para solucionar y evaluar el problema de la selección de métricas de calidad de un software, considerando algunas técnicas para la recopilación de las ideas que puedan dar soporte a la elaboración de una propuesta de solución al problema que en este caso es el aseguramiento de la calidad basadas en las métricas de evaluación definidas en los estándares actuales conocidos. Todo esto es posible gracias al apoyo de herramientas web que proporcionan plantillas para abarcar las fases del Design Thinking. También se hace el detalle de las fases empezando por la fase de Empatizar seguido de Definir, Idear, Prototipar, Testear y finalmente llegando a los resultados y las conclusiones encontradas luego de todo el proceso del Design Thinking. Se espera que los resultados ayuden a la construcción de software de calidad y sean utilizados en un proyecto real y en sus diferentes etapas haciendo que el software tenga éxito en el mercado.

**Palabras claves:** Design Thinking, Métricas, Calidad, Software, Herramientas, Estándares.



## ABSTRACT

*In this article, the Design Thinking technique will be used to solve and evaluate the problem of the selection of software quality metrics, considering some techniques for the collection of ideas that can support the development of a solution purpose to the problem that in this case is the quality assurance based on the evaluation metrics defined in the current known standards. All this is possible thanks to the support of web tools that provide templates to cover the phases of Design Thinking. The details of the phases are also made starting with the Empathizing phase followed by Define, Devise, Prototype, Test and finally arriving at the results and conclusions found after the whole Design Thinking process. The results are expected to help build quality software and be used in a real project and in its different stages making the software successful in the market.*

**Keywords:** Design Thinking, Metrics, Quality, Software, Tools, Standards.

---

## INTRODUCCIÓN

Nuestra meta es buscar los diferentes conflictos sobre la calidad de software para lo cual se va a evaluar y dar solución a la generación de problemas cuando se elige las métricas de calidad de Software para ello se recopila información acerca de cómo se eligen las métricas de calidad para el desarrollo de software. El escoger una métrica de calidad resulta complicada debido al presupuesto reducido, la documentación puede extenderse demasiado y no se hace seguimiento debido del proceso de desarrollo del software. Es por ello que usar el Design Thinking nos permite identificar problemas que pueden ser resueltos por medio del uso de herramientas modernas, estándares alcanzables en la industria del desarrollo de software.

## Materiales y métodos o Metodología computacional

El Design Thinking es una metodología que permite construir ideas en base a la función y a emociones desde la perspectiva del usuario final. Se generan soluciones que se basan en la idea de ponerse en el lugar del usuario final, es decir para quien va dirigido el producto o servicio.

Se escogió esta metodología de trabajo porque permite ver las situaciones de diferentes partes del proceso de la calidad de software en donde participan diferentes equipos de trabajo cuyos integrantes desempeñan diferentes roles y así se pueden generar tantas ideas como sea posible.



## Empatizar

Para la primera fase del Design Thinking se emplearon las siguientes técnicas: a) MoodBoard, b) Observación Encubierta, c) Que Como y Porque y d) Mapa de Actores. Estas técnicas permitieron identificar problemas desde la perspectiva de un encargado de la calidad en el desarrollo de software puesto que son ellos quienes seleccionan las métricas que mejor convengan al proyecto y que deben ser evaluadas para medir el aseguramiento de la calidad.

## MoodBoard

Un MoodBoard permite plasmar las ideas de un problema o situación de manera gráfica y es puede ser utilizada como apoyo para distinguir los tópicos de los problemas de calidad en nuestro caso.



Figura 1. Mapa mental sobre Design Thinking.



## Observación encubierta

Permite mezclarse con el entorno y captar y evaluar la información obtenida. En el entorno de desarrollo de software la elección de métricas muchas veces no se da por cuestiones de tiempo, o requerimientos inconclusos, en las empresas pequeñas es más difícil ver la rama de ingeniería de software porque no se cuenta con el personal ni el presupuesto adecuado para asegurar la calidad, dando más prioridad al análisis y al desarrollo desordenado.

## ¿Qué?, ¿Cómo? y ¿Por qué?

Tabla 1.¿ Resumen del Qué Cómo Por Qué?

¿Qué está Pasando?	¿Como está Pasando?	¿Por qué está pasando?
Las métricas no se gestionan correctamente.	El personal encargado no está usando formatos correctos para evaluar la calidad.	Por que no se entendió correctamente la métrica para elaborar formatos de evaluación de calidad.
No se usan herramientas modernas de gestión de métricas de calidad.	Se hace uso excesivo de documentación que se incrementa y no se maneja controla correctamente.	Las herramientas que existen para manejar documentación son caras.
Se eligen las mismas métricas de calidad para varios proyectos de desarrollo.	En los proyectos de desarrollo se utilizan métricas que se han usado en otros proyectos y se evalúan de la misma manera como lo hicieron anteriormente.	Las métricas de calidad se pueden desglosar en diferentes características que complican entender completamente la métrica de esta forma el personal no logra aplicar nuevas métricas por que creen que ya es suficiente con las que manejan.

## Mapa de Actores

Listado de actores

Personas:

- Jefe de Proyecto.
- Programador.
- Diseñador.
- Usuario.
- Analista.
- Probador(tester).



Organizaciones:

- Empresa de software.
- Empresa distribuidora.
- Empresa usuario.

Instituciones

- Gobierno

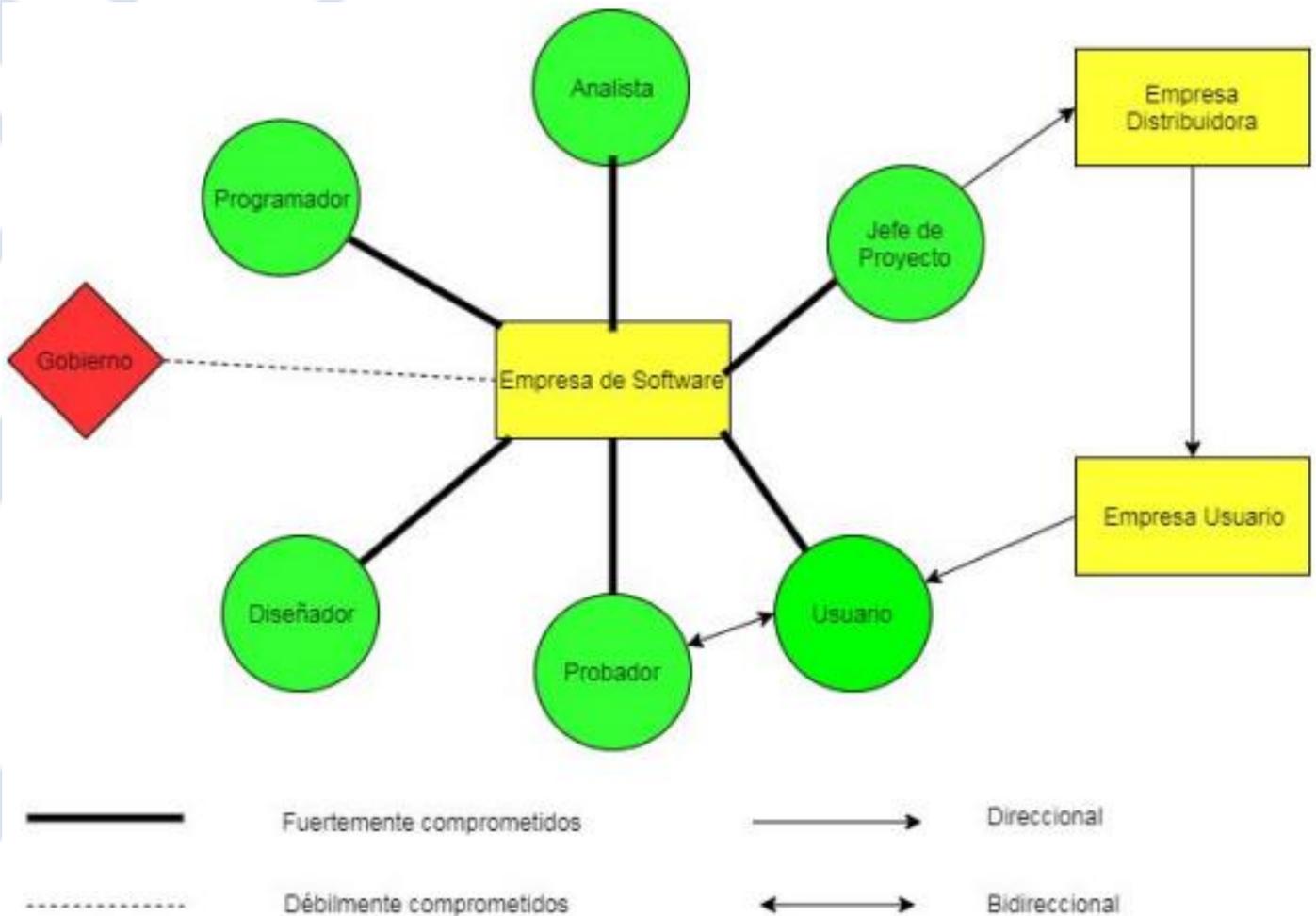


Figura 2. Mapa de actores

## Definir

Para esta fase se usaron las siguientes técnicas: Un mapa mental, Points Views y Blue Print. El objetivo de esta fase es identificar las posibles mejoras que podemos proponer en la selección de



métricas de la calidad partiendo de situaciones que no se desarrollan bien sin embargo estas deben ser alcanzables.

## Mapa Mental

Realizar mapas mentales permite poder ordenar las ideas encontradas acerca del problema a resolver de la mejor manera entendible y poder interpretar conceptos.



Figura 3. Mapa mental



## Points Views

Mediante la técnica de Puntos de vista es uno de los pilares para construir ideas, para crear se usa la palabra necesita y por qué.

Se identificaron los siguientes points views.

- La empresa necesita asegurar la calidad de software porque le da más valor al producto.
- La empresa necesita usar métricas de software porque le da más valor al producto.
- La empresa necesita personal adecuado en calidad de software porque aumentará la calidad del software.
- Las empresas necesitan un mejor proceso de desarrollo porque eso aumentaría su competitividad.
- El analista necesita que se gestione correctamente las métricas de software porque tiene que elaborar mejores formatos para evaluar la calidad de software.
- El analista necesita mejores formatos de software porque la documentación creciente es difícil de manejar.
- El analista necesita mejores herramientas porque eso facilita su trabajo.
- La empresa necesita gestionar elegir las métricas correctas porque eso aumentará la calidad del proceso de software.
- Las empresas usuario necesita un correcto proceso de desarrollo del software porque necesita asegurar que le entrega un buen producto.

## Blue Print

El blueprint que se presenta a continuación identifica procesos que se dan en el desarrollo de software que están directamente relacionados con las métricas de la calidad.

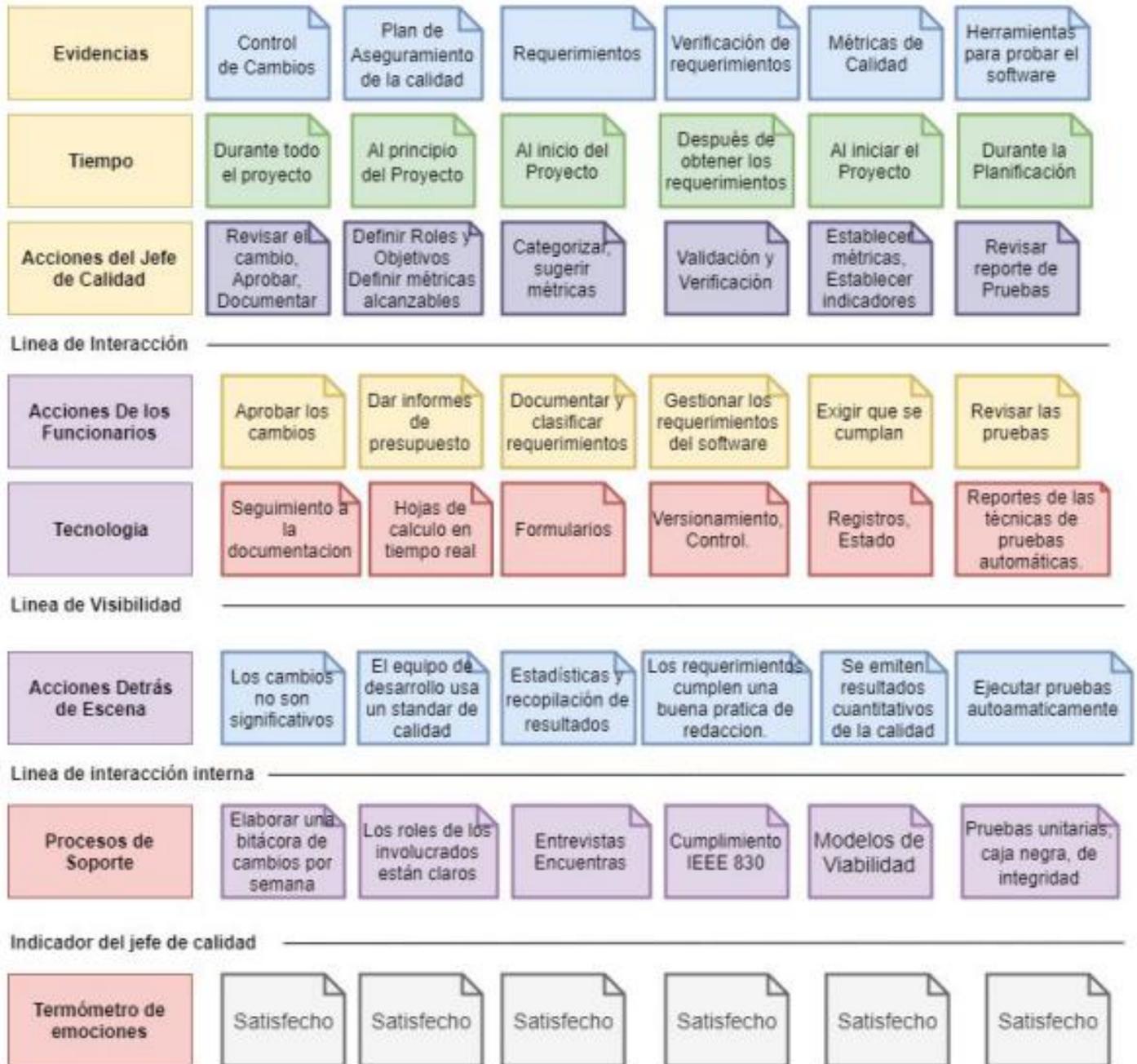


Figura 4. Diseño del Blue Print.

El blue print determina que la documentación de un proyecto se extiende en el tiempo y hace falta encontrar alguna herramienta o definir procesos de revisión y seguimiento.



También que se requiere utilizar un estándar para la mejorar la calidad de los requerimientos puesto que deben de ser entendidos por todos los involucrados y en esa situación es necesario mantener buenas prácticas de documentación de requerimientos para que se mantenga la calidad en el proceso. También se ve la necesidad de manejar un control sobre el resultado de las pruebas además sería de gran utilidad tener automatizada la realización de las pruebas para evitar una sobrecarga de trabajo. Esto tiene incidencia en la calidad ya que los resultados de las pruebas se pueden utilizar como entradas para las métricas de la calidad del software.

## Idear

Permite definir los conceptos esenciales a tratar ideas simples de entender y que son los elementos conceptuales que surgen del grupo encargado de resolver el problema en base a lo aprendido o saberes previos.

## Lluvia de Ideas



Figura 5. BrainStorm.



Con esta técnica de Se puede desprender que se deberían tomar las siguientes acciones para resolver el problema de las métricas de calidad:

1. Manejar un estándar para recoger requerimientos.
2. Revisar que los requerimientos estén completos.
3. Capacitaciones a los involucrados en la calidad.
4. Manejar formatos definidos por el usuario al inicio del proyecto.
5. Incluir que se busca conseguir usando la métrica en el proyecto.
6. Construir una herramienta para gestión de métricas.
7. Usar un estándar de calidad alcanzable.
8. Manejar un seguimiento a la documentación de forma automática.
9. Definir los roles al inicio y de forma clara.
10. Definir qué pruebas se van a realizar.
11. Realizar reportes de pruebas.
12. Gestionar un buen grupo personal para afrontar los proyectos futuros.
13. Hacer un estudio para ver la profundidad del problema y envase a eso tomar las medidas preventivas.
14. Crear reglas y política de comunicación realistas.
15. Apoyo en las tecnologías para automatizar y disminuir el tiempo de estos procesos.
16. Apoyo en las tecnologías para buscar métricas realistas.
17. Buscar normas que se ajusten al problema.
18. Planear un buen recurso del tiempo y gestión de este.
19. Monitorear constantemente si las métricas son las correctas.



20. Tener una bitácora de los errores para posteriormente corregirlos.
21. Fundamentar las métricas usadas en el software.
22. Informar constantemente a los miembros del equipo sobre el funcionamiento correcto de las métricas.

## Prototipar

Nosotros proponemos que se use una herramienta que haga seguimiento al cumplimiento de las métricas de calidad permitiendo al encargado de la calidad tener un control de toda la documentación necesaria para evitar tener que realizar búsquedas y correcciones que no se necesita hacer.

Asimismo, se podrán gestionar diferentes métricas de calidad de estándares de calidad que sean alcanzables en el tiempo así las métricas pueden ser comprensibles por todo el equipo de trabajo encargado de la calidad del software sin que en los proyectos se usen los mismos estándares de calidad. IEEE 830 , ISO 9001, ISO 9126.

Definir roles y responsabilidades desde el inicio del proyecto en cuanto a la calidad por lo que se deberían manejar perfiles de trabajo para el aseguramiento de la calidad del producto donde se describa sus funciones y las comunicaciones que debe manejar para cumplir con sus labores. Por Ejemplo: Perfil del responsable de la calidad:

- Realiza tareas de trabajo con procedimientos e instrucciones de trabajo relativas a los requisitos del estándar seleccionado para el proyecto y de seguridad.
- Actúa como enlace entre los analistas y programadores para asegurar la calidad del producto.
- Selecciona las métricas de calidad para un proyecto, hace seguimiento a la documentación elabora los formatos de trabajo y entrega de resultados de calidad del software.
- Gestiona todos los procesos de calidad designa responsabilidades y revisa el cumplimiento de las métricas de calidad.
- Selecciona herramientas de seguimiento de documentación y la emisión de los informes de estado correspondientes.
- Realiza seguimiento a la ejecución de las pruebas y es encargado de aprobar los reportes correspondientes.



Definir medios de comunicación en la nube para el desarrollo del proyecto de forma que se puedan reducir costos así mismo se podrían establecer formatos que puedan ser usados durante todas las fases del desarrollo y que puedan ser reusadas y modificadas en otros proyectos evitando así hacer uso excesivo de papel así mismo automatizar el proceso de realización de pruebas y la emisión de los reportes correspondientes.

Usar Herramientas modernas de seguimiento de documentación (Gratis o Pago) que se encuentran en la nube. Realizar las pruebas durante todo el ciclo de desarrollo con la inspección del encargado de calidad.

## Testear

Se observó que la herramienta funciona correctamente pero el escoger la herramienta genera dificultad, la instalación y la integración al proyecto en cuestión demora tiempo y al optar por herramientas de pago estas cuestan y en algunos casos piden mensualidad. Se observó que los pasos propuestos funcionan como guía para la realización de un buen aseguramiento de calidad.

## Resultados y discusión

El prototipo creado servirá para la mejora de la calidad en empresa, o grupos de desarrollo de software. Las pautas anteriores a vista de un desarrollador son las mejores elecciones para mejorar la calidad de software. viendo que soluciona parcialmente la déficit de calidad y elección de métricas del software.

También en el prototipo de tendría dificultad a la hora de encontrar una herramienta o grupo de herramientas para gestionar la calidad al existir muchas en el mercado y alguna gratuidad y de pago. siendo en algunos casos complicado la instalación y en otros el coste.

El uso de la técnica Design Thinking te da normas más claras a la hora de generar un buen protocolo de aseguramiento a la hora de elegir métricas de calidad. Al ser basado en prototipos se minimiza el costo y riesgo del fracaso del prototipo.

La herramienta llamada mapa de Actores de la fase de empatizar nos da una visión general de cómo es que actúan y se relacionan los actores en un proceso normal de aplicación de calidad de software en una empresa de desarrollo, nos ayuda mucho para detectar problemas que pueden existir dentro del equipo de desarrollo, posibles cuellos de botella en el flujo de información; Posteriormente a la realización de esta técnica se definen estrategias con las redes sociales que existen y se fortalecen las relaciones que se requieran.



## CONCLUSIONES

Usar la técnica de Design Thinking facilitó el encontrar dificultades en el manejo de la documentación del proyecto durante las fases de empatizar y definir por que la documentación se da durante todo el desarrollo y le es de utilidad a todos los involucrados ya que se pueden dar cambios que no se pueden documentar de la forma correcta sin tener procedimientos claros definidos y los encargados de esta actividad ven barreras que no pueden superar en el proyecto.

El paso a seguir generado por Design Thinking se considera buenas prácticas de calidad a la hora de desarrollar software. Design Thinking ayuda a explorar más ideas y más rápido de lo que podrías utilizando otro concepto, adopta un enfoque centrado en el ser humano con consideraciones comerciales y tecnológicas, la innovación debe tener en cuenta el comportamiento humano, las necesidades y preferencias.

Los resultados de la aplicación de Design Thinking son mejores si se usan templates o plantillas de trabajo que que se encuentran en la web y realizar las fases con métodos tradicionales es un poco tedioso, así que es indispensable valerse de alguna herramienta web para realizar todos los diseños que son requeridos.

## REFERENCIAS

- [1] Brown, Tim. "Design thinking." Harvard business review 86.6 (2008): 84. Available: <https://fusesocial.ca/wp-content/uploads/sites/2/2018/06/Design-Thinking.pdf> [Accessed November 01, 2019].
- [2] Herbsleb, James, et al. "Software quality and the capability maturity model." Communications of the ACM 40.6 (1997): 30-40. Available: <https://pdfs.semanticscholar.org/90cd/732b24707975d7adde20a58c302f23e3dc14.pdf> [Accessed November 01, 2019].
- [3] Slaughter, Sandra A., Donald E. Harter, and Mayuram S. Krishnan. "Evaluating the cost of software quality." Communications of the ACM 41.8 (1998): 67-73. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.450.3664&rep=rep1&type=pdf> [Accessed November 01, 2019].
- [4] Haag, Stephen, MmK Raja, and Lawrence L. Schkade. "Quality function deployment usage in software development." Communications of the ACM 39.1 (1996): 41-49. Available: [https://www.researchgate.net/profile/Stephen\\_Haag/publication/220420894\\_Quality\\_Function\\_Deployment\\_Usage\\_in\\_Software\\_Development/links/53f4b5de0cf2888a749114b7/Quality-Function-Deployment-Usage-in-Software-Development.pdf](https://www.researchgate.net/profile/Stephen_Haag/publication/220420894_Quality_Function_Deployment_Usage_in_Software_Development/links/53f4b5de0cf2888a749114b7/Quality-Function-Deployment-Usage-in-Software-Development.pdf) [Accessed November 01, 2019].



- [5] Sánchez, Ana María García, and Dtor Manuel García Clavel. "Evaluación de métricas de calidad del software sobre un programa Java." Universidad Complutense de Madrid, Madrid (2010). Available: [https://eprints.ucm.es/11487/1/Proyecto\\_Fin\\_de\\_M%C3%A1ster.pdf](https://eprints.ucm.es/11487/1/Proyecto_Fin_de_M%C3%A1ster.pdf) [Accessed November 01, 2019].
- [6] Müller, Roland M., and Katja Thoring. "Design thinking vs. lean startup: A comparison of two user-driven innovation strategies." *Leading through design* 151 (2012): 91-106. Available: [https://s3.amazonaws.com/academia.edu.documents/40692530/Leading\\_Innovation\\_through\\_Design\\_Proceedings20151208-21966-ccwlds.pdf?response-content-disposition=inline%3B%20filename%3DLeading\\_Innovation\\_through\\_Design\\_Proceedings20151208-21966-ccwlds.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20191101%2Fus-east-1%2Faws4\\_request&X-Amz-Date=20191101T234613Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=e472010c8014e117a49f4a85a6f0363cd5436b9f4a05ddfb08adbc0ef4b3c868#page=181](https://s3.amazonaws.com/academia.edu.documents/40692530/Leading_Innovation_through_Design_Proceedings20151208-21966-ccwlds.pdf?response-content-disposition=inline%3B%20filename%3DLeading_Innovation_through_Design_Proceedings20151208-21966-ccwlds.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20191101%2Fus-east-1%2Faws4_request&X-Amz-Date=20191101T234613Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=e472010c8014e117a49f4a85a6f0363cd5436b9f4a05ddfb08adbc0ef4b3c868#page=181) [Accessed November 01, 2019].
- [7] Gaffney Jr, John E. "Metrics in software quality assurance." *Proceedings of the ACM'81 conference*. ACM, 1981. Available: [https://people.dsv.su.se/~joco2917/ft\\_gateway.102cfm.pdf](https://people.dsv.su.se/~joco2917/ft_gateway.102cfm.pdf) [Accessed November 01, 2019].
- [8] Caldiera, Victor R. Basili, Gianluigi, and H. Dieter Rombach. "The goal question metric approach." *Encyclopedia of software engineering* (1994): 528-532. Available: [https://s3.amazonaws.com/academia.edu.documents/40605563/gqm.pdf?response-content-disposition=inline%3B%20filename%3DThe\\_Goal\\_Question\\_Metric\\_Approach.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20191101%2Fus-east-1%2Faws4\\_request&X-Amz-Date=20191101T234948Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=1de103e59d63c998e9d0ecd4f369abbc63aa69cfb9be9325e38d90a051474b3a](https://s3.amazonaws.com/academia.edu.documents/40605563/gqm.pdf?response-content-disposition=inline%3B%20filename%3DThe_Goal_Question_Metric_Approach.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20191101%2Fus-east-1%2Faws4_request&X-Amz-Date=20191101T234948Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=1de103e59d63c998e9d0ecd4f369abbc63aa69cfb9be9325e38d90a051474b3a) [Accessed November 01, 2019].
- [9] Kolko, Jon. "Design thinking comes of age." (2015): 66-71. Available: [https://enterpriseproject.com/sites/default/files/design\\_thinking\\_comes\\_of\\_age.pdf](https://enterpriseproject.com/sites/default/files/design_thinking_comes_of_age.pdf) [Accessed November 01, 2019].
- [10] Steinbeck, Reinhold. "El «Design Thinking» como estrategia de creatividad en la distancia." *Comunicar* 19.37 (2011): 27-35. Available: <https://www.redalyc.org/pdf/158/15820024004.pdf> [Accessed November 01, 2019].



# Artificial Intelligence in Digital Health: Issues and Dimensions of Ethical Concerns

81

*Inteligencia Artificial en Salud Digital: Cuestiones y dimensiones de los aspectos éticos*

**Fredrick R. Ishengoma**

The University of Dodoma. Dodoma, United Republic of Tanzania.

 [ishengomaf@gmail.com](mailto:ishengomaf@gmail.com)

 <https://orcid.org/0000-0002-3337-6210>

 **ARK:** [ark:/42411/s8/a48](https://nbn-resolving.org/urn:nbn:org:ark:/42411/s8/a48)

 **PURL:** [42411/s8/a48](https://nbn-resolving.org/urn:nbn:org:ark:/42411/s8/a48)

RECIBIDO 15/12/2021 • ACEPTADO 08/02/2022 • PUBLICADO 30/03/2022

## ABSTRACT

Artificial intelligence (AI) is transforming the healthcare system at a breakneck pace by improving digital healthcare services, research, and performance, fueled by the combination of big data and strong machine learning algorithms. As a result, AI applications are being employed in digital healthcare domains of which some were previously regarded as only done by human expertise. However, despite AI's benefits in digital healthcare services, issues and ethical concerns need to be addressed. Using mapping review methodology, a taxonomy of issues and ethical concerns surrounding the employment of AI in healthcare is presented and discussed. Moreover, policy recommendations and future research directions are presented.

**Keywords:** Artificial Intelligence, Digital healthcare, Ethics, Machine Learning.

## RESUMEN

*La inteligencia artificial (IA) está transformando el sistema de atención médica a un ritmo vertiginoso al mejorar los servicios, la investigación y el rendimiento de la atención médica digital, impulsados por la combinación de grandes datos y sólidos algoritmos de aprendizaje automático. Como resultado, las aplicaciones de IA se están empleando en dominios de atención médica digital, algunos de los cuales antes se consideraban realizados solo por experiencia humana. Sin embargo, a pesar de los beneficios de la IA en los servicios de atención médica digital, es necesario abordar los problemas y las preocupaciones éticas. Utilizando la metodología de revisión de mapeo, se presenta y discute una taxonomía de problemas y preocupaciones éticas que rodean*



*el empleo de la IA en el cuidado de la salud. Además, se presentan recomendaciones de política y direcciones de investigación futuras.*

**Palabras claves:** *Inteligencia artificial, Sanidad digital, Ética, Machine Learning.*

---

## INTRODUCTION

Artificial Intelligence (AI) is bringing a fundamental change in digital healthcare services, thanks to the growing availability, accessibility of data and the rapid advancement of advanced analytics [1,2]. The significant growth of digital data, the advancement of computational power bolstered by innovation in hardware, including graphics processing units and machine learning (ML) techniques, widely applied using deep learning (DL), are all creating an indelible mark in the healthcare domain [3]. This has attracted attention, and a considerable number of research on the effective usage of ML in big volumes of health data [4-6]. Moreover, the increased AI use is sought to lower the significantly higher rate of human error in healthcare services, which is likely to result to injury or even death [7]. For instance, scientists are researching the effective use of DL to imitate the neural networks in the human brain and are being tested in hospitals by businesses like Google to see if these machines can help with decision-making by anticipating what will occur to a patient [8]. However, several ethical concerns arise in the healthcare domain when AI systems perform the same tasks as humans [9-12]. For instance, consider a scenario when the robot commits a calculating error and prescribes the incorrect dose of medicine, resulting in catastrophic harm or death. Additional queries also arise in the same vein, such as what if, on the other hand, AI machines result in new kinds of medical errors? And who will be found liable if they occur? [14, 15]. In such a setting, some scholars argue that policies and rules for ethical AI in health care should be inevitable to these rising challenges [13].

In the meantime, policies and ethical guidelines for healthcare services that use AI and its implementations lag behind the pace of AI advancements [16]. As a result, existing AI-based technologies and applications must be examined and discussed to address ethical concerns. Thus, in this paper, the existing issues and ethical concerns in the employment of AI in digital health are presented and discussed. The paper is structured as follows: Section 2 presents the overview of AI employment in digital health. Section 3 presents the methodology used in this study. Section 4 discusses the existing issues in employing AI in digital healthcare services. Section 5 discusses and presents the ethical dimensions of AI concerns in digital healthcare along with the taxonomy. Finally, the paper is concluded in section 5 with conclusions, policy recommendations and future research directions.



## AI Employment in Digital Health

In digital healthcare (DH) services, AI can be employed under virtual and physical categories. The virtual portion includes viewpoints from electronic and system perspectives such as Electronic Health Record (HER) systems, Natural Language Processing (NLP), expert systems, to neural network-based treatment decision assistance [17]. The physical section covers themes such as robotic surgery assistants, intelligent prosthetics for disabled persons, and senior care [18].

Robots are becoming more collaborative with people, and they are easier to accommodate by guiding them through a task, and they are also becoming smarter as more AI functionalities are integrated into their 'brains' [18]. The same advancements in intelligence that we have seen in other AI fields will soon be relevant to physical healthcare robots. For example, surgical robots allow surgeons to see better, make more precise and least invasive incisions, repair wounds, and so on [19]. Medical experts can now care for a higher number of patients by using AI. AI tools can assist them in making better diagnostic judgments, improving treatment outcomes, and reducing medical errors. AI could also help with HR difficulties like recruitment and selection of potential healthcare workers [20].

The most crucial question, meanwhile, remains an open question: "Are we willing to give life and death choices to AI?" "Can computers definitively determine whether or not the treatment given to a patient is adequate?" Addressing the concerns above is part of the ongoing research and may be challenging due to the multiple hurdles and difficulties that AI and robotics may entail. Nevertheless, one thing is certain: AI and robotics will continue to play a significant role in DH services.

In the meantime, there has been a significant growing amount of data available for assessing healthcare activity and biological data in recent years. With the rising amount of data, DL is being applied in figuring disease patterns such as cancer in early stages, thanks to the ongoing advances in processing power [21]. In addition, consumer wearables and other medical equipment, blended with AI, are being used to identify and detect possible life episodes in initial heart disease, allowing doctors and other carers to supervise better and detect possibly serious incidents sooner, more fixable phase. Thus, pattern recognition is being used to identify people at risk of getting an illness – or seeing one worsen – due to lifestyle, environmental, genetic, or other variables. EMR databases store information about previous hospital visits, diagnoses and treatments, lab results, medical photographs, and clinical narratives. These datasets can be used to create prediction models to assist physicians with diagnoses and treatment decision-making. As AI techniques improve, it will be feasible to extract a wide range of data, including disease-related impacts and connections between past and future medical events. Thus, even though AI applications for EMRs are presently restricted, the potential for employing huge datasets to discover new patterns and forecast health consequences is immense [23].



Furthermore, one of the more modern uses of AI in healthcare is drug research and development. There is the possibility to drastically reduce both the time to market for new pharmaceuticals and their prices by directing the current breakthroughs in AI to expedite the drug discovery and drug repurposing processes. In a manner that basic computer-driven algorithms cannot, AI allows individuals in training to be in a realistic simulation.

Precision medicine is another common use of classical AI in DH, which involves forecasting which treatment protocols are most likely to succeed on a patient depending on various patient characteristics and the treating context [22]. Precision medicine solutions employ ML mostly under supervised learning, which requires a training dataset with a known outcome measure (e.g., illness onset).

However, despite the advantages of AI in DH, several limitations still exist. For instance, computers may interpret human behaviour rationally and logically, but many human characteristics like critical thinking, interpersonal and communication skills, emotional intelligence, and imagination are not refined by computers. Thus, AI has gone a long way in the healthcare profession, yet human supervision is still required. Surgery robots, for instance, work rationally rather than empathically. Nevertheless, health practitioners may see important behavioural insights that might assist identify or avoiding medical issues.

Furthermore, AI systems may be blind to social factors [44]. For example, an AI system could assign a patient to a certain treatment centre based on a diagnosis. However, this approach may not take into consideration the patient's financial constraints or other personal preferences. Medical AI is primarily reliant on diagnostic data culled from millions of instances. In circumstances where there is a scarcity of information on a certain illness, misdiagnosis is likely to happen [45].

## Methodology

The mapping review was conducted to map the literature in the issue area and assist in determining the study's eventual scope. Methods aided in identifying the literature described by [43]. The author examined academic resources such as Springer, Elsevier, Taylor and Francis, IEEE, PubMed, MEDLINE and Wiley for "AI and healthcare services" published between November 2020 and November 2021.

Titles and abstracts were checked, and full-text papers were reviewed if they were appropriate. Quantitative analysis and a major focus on AI and healthcare services, especially on issues and ethical considerations, were the inclusion requirements. Non-English and non-AI studies were both ruled out.



Significant studies were additionally snowballed by citation searching in Google Scholar, and reference lists included in publications were inspected. Next, the identified references were evaluated by a group of three reviewers. To ensure uniformity, the first 100 references were examined by all three reviewers. Finally, any questions were answered by talking with the other two reviewers. After the title and abstract screening, the references that could meet the inclusion criteria were given a second look, and data was retrieved for use in the review. Citations that did not match the inclusion criteria were removed from the study.

For mapping reasons, references were divided into sets as per AI issues in healthcare and AI ethical concerns in digital healthcare services. Data from each abstract was used to complete data extraction. If an abstract was not accessible, minimal details from the title were collected for the mapping review, understanding that the full text would be retrieved if the mapping review was included in the intended systematic review.

## AI Issues in Digital Health

The advancement of artificial intelligence in healthcare is fraught with issues and challenges. Errors in AI systems, for example, put patients at risk of harm. Similarly, using a patient's data for AI research puts the patient's privacy in danger. In this section, issues of employing AI in digital health are discussed.

## Training Data

To correctly train neural networks and guarantee an effective neural network algorithm in AI, huge volumes of data are required (citation). Systems that are robust and accurate cannot be designed with too little data [24,25]. As a result, overfitting<sup>1</sup> can occur, resulting in data that does not generalize well enough to new data.

Overfitting happens when data is trained too well, lacking the ability to fit the assessment data and resulting in poor results. Furthermore, several governments may find it challenging to use AI since they lack the necessary data to train neural networks.

Deep learning also relies significantly on labelled data to ensure that the algorithm's output is high quality. As a result, these datasets necessitate a large number of experts and data analysts. For some rare health disorders, open-source data is available; however, data on more infectious ailments, on the other hand, is limited [26].

---

<sup>1</sup> Overfitting happens when data is trained too well, lacking the ability to fit the assessment data and resulting in poor results (citation).



Presently AI systems are heavily reliant on their training data; as a result, these algorithms' accuracy is limited by the information in the datasets on which they are trained, which means they can't escape biases and errors in the training data. Furthermore, due to differences in patient demography, physician preferences, equipment and resources, and health policies, medical information obtained in clinical practice often varies among entities and contexts [27].

This brings us to the limitation of data usage according to the pattern and context of its collection. Thus, for instance, can healthcare professionals in developing countries, for instance, depend on AI decision-making capabilities that have grown from data from developed countries based on their group of the population that is distinct from the developing countries population?

Moreover, much data is unstructured and unorganized regarding their final form and gathering method, with missing data regularly occurring [28]. As a result, the vast majority of clinical data are inadequate for AI algorithms to use effectively.

In the meantime, producing and annotating this much medical data takes a lot of time and effort. This has led to few available databases on which its hidden biases in such information may not be apparent. As a result, researchers collecting large amounts of medical data to construct AI systems may rely on whichever data is available, even if it is subject to numerous selection biases.

## Blackbox and Explainable AI (XAI)

The "black box" hinders the ability to see how the algorithms work within AI-based healthcare systems. The black box makes decisions based on many connections, making it impossible for the human mind to understand how and on what basis the decision was made [29]. As a result, it calls into question the integrity of the data and its modus operand.

This means that neither doctors nor patients can understand how the AI system arrived at its choice [30]. Therefore, transparency is required in AI-based healthcare systems to perform proper and effective system evaluations and audits. Thus, an AI-based healthcare system must be auditable, whereby transparency should include correct information on the technology's premises, constraints, operating protocols, data attributes (including data collecting, processing, and labelling methods), and algorithmic modelling. In addition, experts have emphasized that explainability is required when an AI offers health recommendations, particularly to uncover biases, in the case of "black-box" algorithms [31]. This has led to the rise of interest in the AI sub-discipline called "Explainable AI"<sup>2</sup> (XAI).

---

<sup>2</sup> Explainable AI (XAI) is a suite of techniques and frameworks that can assist users comprehend and understand machine learning predictions.



The XAI extends to how AI machines can know the context and environment in which they work and construct appropriate explainable models that allow AI to define important considerations across time Figure 1.

XAI can improve the usability of AI-based digital healthcare services by assisting end-users in trusting that the AI makes smart decisions which is essential in digital healthcare services. The goal of XAI in this manner is to convey what has been undertaken, data, to reveal the knowledge on which the actions are based and how the AI come to make a decision [32].

This argument is currently under discussion in the academic arena, whereby some scholars believe that what counts is that the AI is correct, at least in the context of diagnosis, rather than how it makes its conclusion [33]. Positive outcomes of randomized clinical trials could be used to establish the safety and usefulness of "black box" health AI applications, comparable to how pharmaceuticals are handled.

## AI Malfunctions

Even with direct guided robotic surgery, robotic faults during surgery are still occurring [34]. For example, consider robotically assisted surgical devices (RASDs), allowing surgeons to manipulate tiny cutting instruments rather than traditional scalpels. If a surgeon's hand slips with a scalpel and a key tendon is sliced, our instinct is that the surgeon is to blame. But what if the surgeon is employing a RASD that is touted as having a unique "tendon avoidance subroutine," comparable to the warnings that cars currently emit when sensors detect a potential collision? Can the wounded patient sue the RASD vendor if the tendon sensors fail and the warning does not sound before an incorrect cut is made? Or was it only the doctor who relied on it?

Thus, surgical errors made by unsupervised robotic surgical devices will certainly be one of the largest legal difficulties in the future, even though there are still significant challenges in direct-control robotic surgery [35]. Of course, a human doctor might have averted the surgical blunders produced by autonomous robotic surgical devices, but these systems may outgrow the need for people in the future.

This raises the following questions: should we keep improving these technologies until the surgery error rate caused by robots reaches zero, and should we continue to allow patient damage due to human error until the system is perfected? Should we promote autonomous robotic surgery after obtaining satisfactory outcomes at the expense of a few patients?

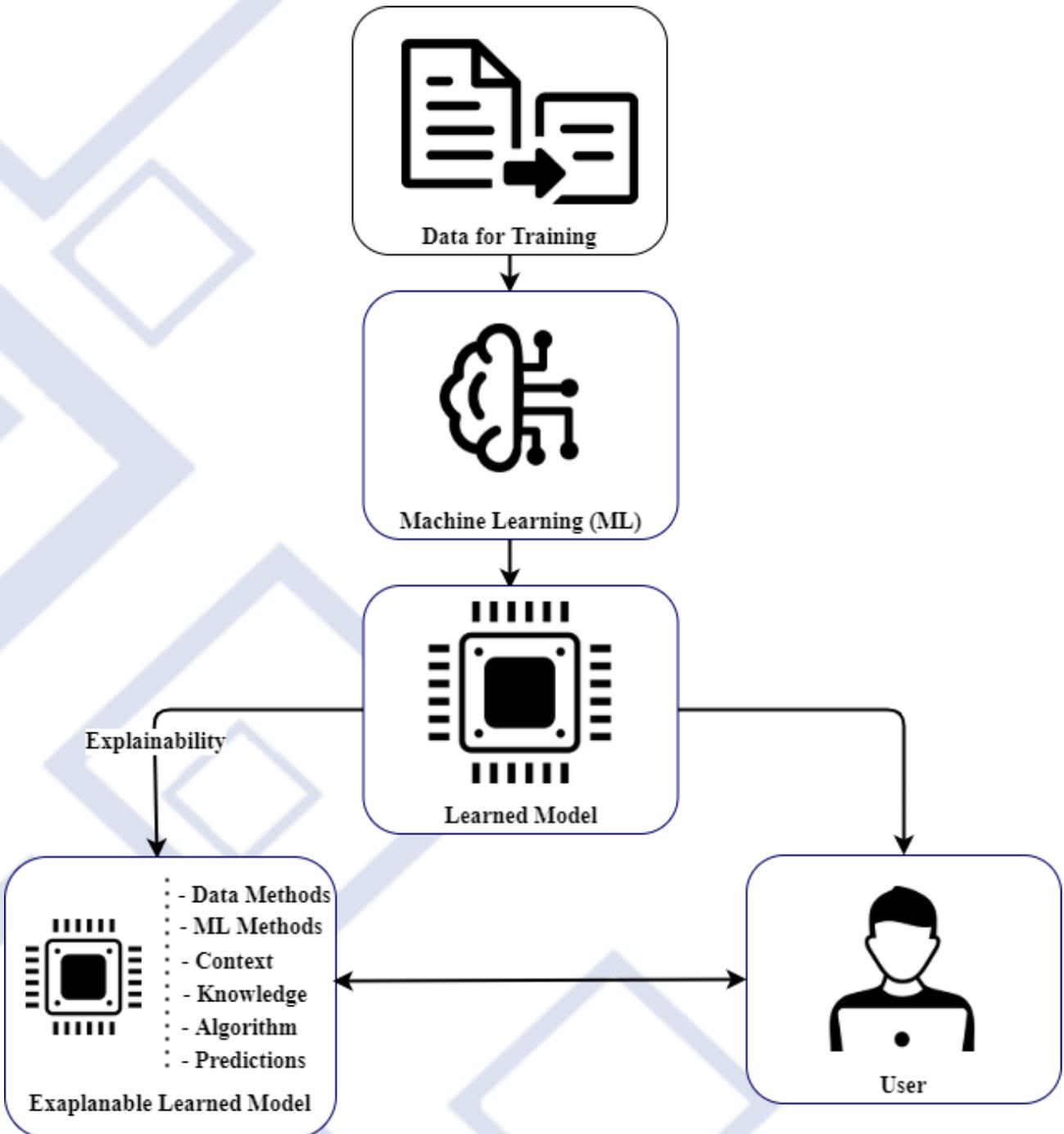


Figure 1: Typical XAI Layout



To answer these problems, further education in the field of AI technology is required. Concerns have been raised about how machine errors could be spotted again. The proponents of AI will give a specific example of AI in airline autopilot mode, which does not jeopardize pilot training but, in my viewpoint, is accountable for autopilot malfunctions that result in plane crashes.

## AI vs Professional Skills

There is a growing concern in academia that the increasing reliance on algorithms will impair people's ability to think for themselves in the long run. As a result, AI is bound to gradually deprive our brains of mental effort and thinking as we become more accustomed to using it in everyday chores. Moreover, according to the researchers, high dependence on automation can erode professional abilities. Thus, as healthcare workers are extensively using AI it impedes the development of doctors' abilities and clinical procedures [36].

Thus, there is the potential deskilling of healthcare professionals due to increased independence in AI. However, one of the ways to tackle this is to employ the human-AI combinatory approach. For example, when it comes to cancer diagnosis, clinicians must be both sensitive and specific to avoid over-flagging of questionable tissue. While humans aren't very sensitive, algorithms are. Therefore, combining the two sets of talents might have a huge positive impact on healthcare.

## Standardization of AI Algorithms

Several researchers throughout the world are developing AI algorithms in DH. Simultaneously, several governments and corporations invest much in AI research on DH. It's possible to wonder if AI research has already resulted in standardized algorithms for digital healthcare services [46]. For AI in DH, standardization work is essential, helpful, and instructive. It represents both a vital lever for driving industrial innovation and the pinnacle of the competitive landscape. While AI-related supplies in DH are becoming more widely available in China, issues with insufficient levels of standardization are also emerging.

Currently, since each study's performance is presented using various approaches on diverse communities with distinct sample distributions and features, objective assessment of AI algorithms across research is difficult. Therefore, algorithms must be compared on the same independent test set that is generalizable, using the same performance standards, to make fair comparisons for a particular domain, for instance, DH [47].

Clinicians will struggle to determine which algorithm is most suitable to accomplish well for their patients if this isn't done. Every healthcare provider's curating of independent local test sets might be utilized to test the efficiency of several accessible algorithms in a sample group of their



population. These separate test sets must be built using an unenriched representative group and intentionally unavailable data for training algorithms. Before actual testing, an additional local training dataset might be given to allow fine tweaking of algorithms.

The rising availability of huge, accessible datasets will simplify comparison for academics, allowing research to compare their effectiveness cohesively.

## **ETHICAL DIMENSIONS OF AI IN DIGITAL HEALTH**

### **Algorithmic Bias**

While AI applications can eliminate human prejudice and mistake, the data used to train them might reflect and reinforce biases.

Concerns have been voiced concerning the possibility of AI causing prejudice in ways that are concealed or do not correlate with protected by law criteria like gender, race, handicap, and age. In addition, the advantages of AI in healthcare may not be dispersed evenly [37]. Where data is limited or difficult to obtain or render electronically, AI may perform less well.

People with rare medical illnesses and disadvantaged in clinical trials and scientific data, such as Black, Asian, and minority ethnic communities, may be affected. Biased AI may, for example, lead to incorrect diagnosis and make medicines ineffectual for particular population groups, jeopardizing their safety in the health sector, where phenotype- and sometimes genotype-related data is involved [38].

For example, consider AI-based clinical decision support (CDS) system that assists physicians in determining the optimum treatments for skin people with cancer. The algorithm, on the other hand, was primarily trained on Asian patients.

As a result, subgroups for which the dataset was underinclusive, including African Americans, may likely receive less accurate or even erroneous suggestions from AI software.

High information availability and efforts to better collect data from minority communities and clearly describe which populaces the algorithm is or is not fit for may help resolve some of these biases. Nevertheless, there is still the issue of several algorithms being complex and opaque.

As the public's trust grows, so makes the provision of information from a wider range of sources. For example, we already know that some diseases exhibit differently depending on the patient's



ethnic origin. A quick illustration can be seen in an AI program developed to detect malignant moles. The AI will have been educated on a database primarily made up of white skin photos in its early phases, making it less likely to detect malignant patterns on darker skins.

Before artificial intelligence, medical datasets and trials had a longstanding experience of prejudice and underrepresentation of women and persons of diverse races and ethnicities. One way in which the results can be skewed is if the dataset utilized for machine learning does not include enough people of diverse sexes, races and ethnicities, or socioeconomic backgrounds.

COVID-19's unequal impact on some racial and ethnic groups mirrors longstanding racial disparities in scientific science and access and serves as a harsh reflection of the need to reduce prejudice in developing health-research technology [39].

## Privacy and Security

Data power AI. Machines would be unable to learn how to 'think' without it. This is why the privacy of patients' medical data is so important, and it has become a global corporate and government focus. Nevertheless, due to the sheer sensitive nature of patients' medical information, the health industry is heavily vulnerable to cyber assaults. Many people consider data that is confidential and private to be used in AI applications in healthcare. The law regulates these.

Other types of data, such as social media interactions and web search history, that aren't directly related to health state, could be used to provide information about the user's and others' health. Artificial intelligence (AI) could be used to identify cyber-attacks and safeguard healthcare information systems. Nevertheless, AI systems could be hacked to obtain access to sensitive information or inundated with phoney or biased information in ways that are difficult to discern [40].

One of the most pressing issues is integrating AI machine learning into clinical settings with informed permission and balancing patient privacy with AI effectiveness and safety [41]. It also raises the question of when a practitioner must inform a patient that AI is being utilized. Moreover, there is a lack of public awareness of how patient data is used, and both doctors and patients want to learn more about it. Does this pose trust issues: can we trust an application to diagnose us superior to a doctor or on par with us? We must first understand why people are terrified of AI before we can create trust. Rather than rejecting them as Dullards, we should consider their concerns and make them a part of the solution.

Values change from one country to the next, as well as from one corporation to the next. While face recognition technology is widely used in China, individuals in the West are wary about such



surveillance. The gathering, use, assessment, and sharing of patient data has sparked widespread concern about privacy rights since a lack of privacy can harm an individual (for example, future discrimination based on one's health status) or cause a wrong (for example, affecting a person's dignity if sensitive health information is distributed or broadcast to others) [42].

Given that personal medical information is among the most sensitive data types, there are serious ethical questions about how access, management, and usage can alter over time as a self-improving Gets better. In creating regulations in this domain, a focus on patient consent would represent the essential ethical ideals.

Requirements for technologically assisted repeated informed consent for new information use, for instance, would serve to protect patients' privacy. Additionally, the right to withdraw data could be explicitly stated and made simple to execute and generated data could be utilized to replace the data gaps left by these agency-driven withdrawals while avoiding the de-operationalization of AI systems.

## **AI, Social Isolation and Human Touch**

AI mobile applications can empower people to assess their symptoms and when appropriate, take care of themselves. Nevertheless, if AI technologies are utilized to replace professional or relatives time with patients, concern has been raised regarding a loss of human touch and growing social isolation [48]. At its best, AI assists physicians in reaching more patients, reducing administrative burden, and increasing treatment accessibility.

In the worst-case scenario, it forms a physical barrier between the doctor and the patient, stifling empathy and jeopardizing patient trust. Touch is an essential component of the human experience. It fosters personal connections, decodes human emotion, and encourages trust and healing in the healthcare setting. But, in our haste to digitize every aspect of the digital healthcare services, are we skipping this fundamental practice?

Beyond the hospital setting, the patient can operate certain AI-driven diagnostic applications directly on portable devices. In 2017, the FDA authorized the first smart pill, which includes a swallowable sensor that sends a signal to the patient's device once the tablet is eaten, assisting them in adhering to medication [49]. However, from a medical point of view, ethical problems for these types of devices include autonomy, confidentiality, and trustworthiness in the event of technical malfunctions.

Consider a situation in which a patient communicates with their doctor via telemedicine (a field that is rapidly incorporating AI to improve patient relations). There will be times when a doctor



will need to see their patient face to face. It's critical to strike the correct balance between knowing when to employ technology and when to complement with human contacts.

The active research questions in this area includes: What influence will the digital transformation and growing use of AI have on the patient-doctor relationship? How can we take use AI in DH while keeping important values like safety, privacy, security, and trust in mind?

## **Liability and Full delegation of AI in Digital Health**

Even if AI robots and algorithms are extremely sophisticated, mistakes can still happen. Humans, like the machines they train, are likely to make mistakes. The ongoing question is, who is liable for the AI mistakes? The medical centre, the health practitioner, or the algorithm's manufacturer [50].

If physicians can't be held responsible, should the AI system be made responsible? Many academics disagree, claiming that AI lacks the human characteristics required to make moral judgements based on empathy and semantic comprehension. However, the fact that AI systems cannot currently be held responsible for their actions should not deter efforts to instil moral responsibility in them. These arguments need to be addressed.

Moreover, we may be unable to track how decisions are taken when AI algorithms are opaque - which they are in many circumstances. As a result, increased openness among participants must be enforced. Those that create or implement AI systems may face legal consequences, though the details of how that responsibility is regulated and enforced are still under debate.

Another difficulty with liability is that AI systems are always advancing and developing, posing new challenges and unique circumstances, as the case of AI systems capable of generating new AI algorithms. So, what—or who—is to be held responsible when an AI system establishes a fully independent system?

Another decision to be made seems to be whether accountability rules should incentivize practitioners to use AI to inform and verify their clinical judgment or to diverge from their judgment if an algorithm reaches an unanticipated outcome.

If healthcare practitioners are penalized for leaning on AI technology that turn out to be erroneous, they may only utilize the technology to corroborate their judgment. While this may protect them from legal guilt, it may inhibit AI from being used to its greatest potential: to augment rather than validate human judgment.



## Digital Divide

The "digital gap," which refers to unequal access to, usage of, or impact of information and communication technology among various populations, is one obstacle that is argued to impede AI adoption. Even though the cost of digital technology is decreasing, access is still unequal. Researchers argued that two-tiered health care could be one of the significant long-term repercussions on the healthcare delivery system.

Is it possible that a two-tier diagnostic service will arise, with only the richest people having access to human-led interpretation of AI in DH? Or, on the other hand, are only the wealthy granted access to a potentially greater machine-led analysis of test findings or imaging?

Training AI takes a lot of time and energy, as well as a lot of computer resources. Machine learning models are typically only run by wealthy countries and universities with substantial computational power. This becomes a barrier to AI and frontier technology democratization.

## Collective Medical Mind

The so-called "collective medical mind" dilemma addresses the transfer of medical authority from human doctors to algorithms. The danger here is that AI systems used as decision support tools will eventually become central hubs in medical decision-making. In this situation, it's unclear how existing medical ethics concepts (consequentialism, beneficence and non-maleficence, and patient respect) can still be anticipated to play an important role in the patient-doctor interaction that they do now—or may be expected to have in the future.

On the other hand, using AI-powered tools to mediate the doctor-patient connection can radically transform the doctor-patient interaction.

AI may increase interpersonal distance among patients and their doctors, particularly as it permits distant care or communication via robotic assistants. The need to streamline patient care could motivate to adopt such tools, but the flip side is that the patient is becoming more and more isolated, which could have detrimental consequences for health outcomes.

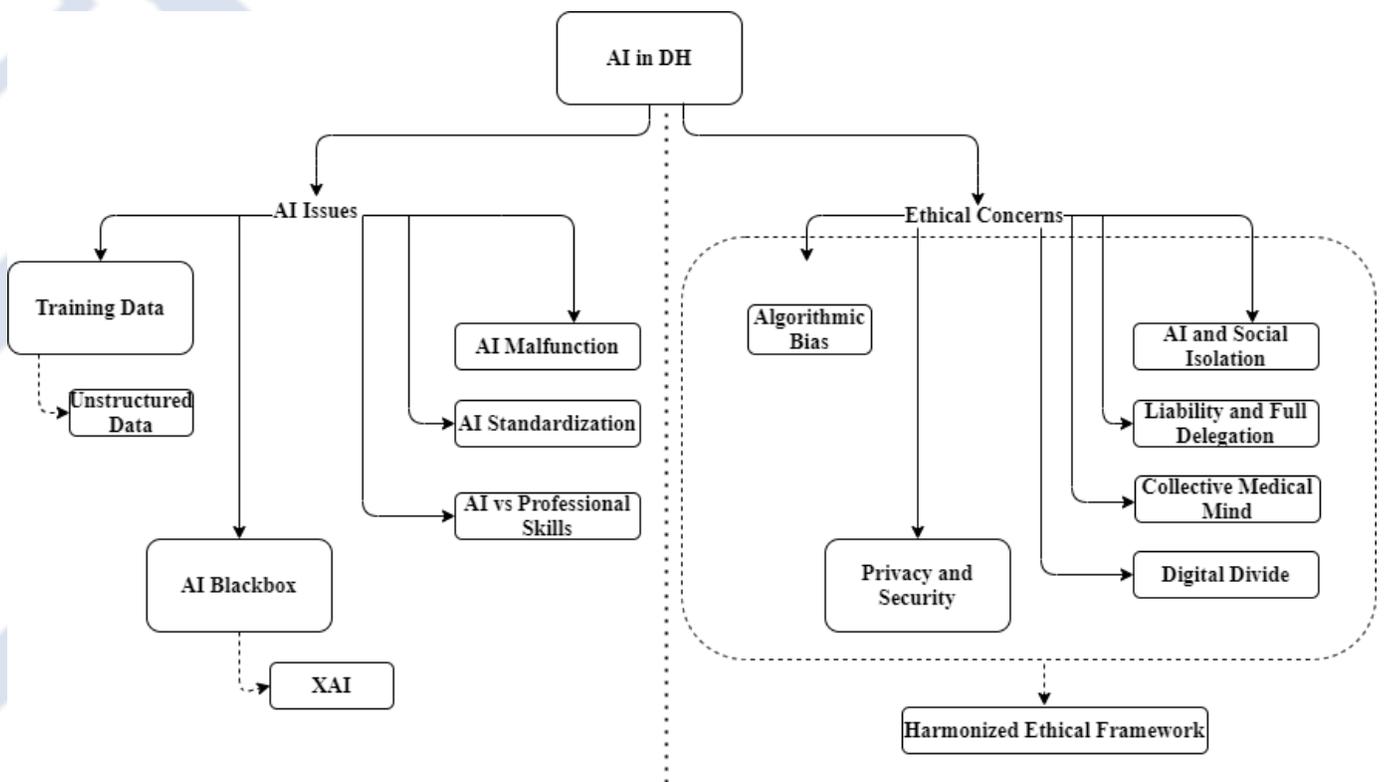
AI-based household platforms are subject to the same concerns. In theory, these technologies might be immensely valuable for providing better care to older people with reduced mobility, for example. They can, nevertheless, exacerbate social isolation.



## Harmonized Ethical Framework

There is little to no relevant multilateral guideline on using AI for health in compliance with ethical principles and provisions currently available. The majority of countries lack legislation or regulations governing the use of AI in health care, and those that do exist may not be sufficient or specialized enough for this purpose.

To generate trust in these technologies and prevent the proliferation of inconsistent norms, ethics guideline based on the common perspectives of the multiple agencies that create, utilize, or oversee such technologies is vital. For the design and deployment of AI for global health, standardized ethics guidelines are required.



**Figure 2:** Taxonomy of AI Limitations and Ethical Dimensions in Digital Health



## CONCLUSION

As AI becomes more integrated into work and personal life, it poses ethical risks in replacing people with robots. These concerns are more pressing in the healthcare field, where decisions can mean life or death. The spread of AI may result in the delivery of health services in unregulated settings and by uncontrolled practitioners, posing issues for government healthcare regulation.

Proper regulatory supervision mechanisms must be devised to guarantee that the private sector is directly answerable to individuals who can gain from AI goods and services and that decision-making and operations are open.

A fundamental problem for future AI governance will ensure that AI is designed and implemented ethically, transparent, and public-interest-compatible while reducing risks and promoting innovation in digital healthcare. Because AI technology draws inferences based on machine learning of the data collected, the decision-making process ignores the unique circumstances of individual patients, raising ethical, moral, and legal concerns.

As a result, it's important to discuss the rules and conventions that AI technology should follow, such as ethics, regulations, and personal beliefs, which govern society's behaviour.

AI is predicted to enable deeper ties between healthcare providers and patients in the long term by, for instance, compensating for AI's faults. As a result, the medical school curriculum should include AI-related learning and technology efficiently. Furthermore, because historical datasets of patients constrain medical AI, more precise AI systems should be created. This progress necessitates the active cooperation of medical professionals at the start of the AI development process.

To tackle this problem, it is vital to cultivate a highly-trained professional team that can react quickly in the consumer-oriented healthcare industry by offering a variety of sophisticated technology learning possibilities, such as leveraging AI to cooperate with medical personnel.

Furthermore, to produce new employment, we believe that new curriculums (for example, technology innovation and application, human-machine confluence, data analytics classes, human-machine exchange, cyber ethics and accountability, and so on) should be incorporated into medical school curricula.

The recommendations in this article are based on existing AI-based technology use, which may limit our comprehension of future technology's full potential. However, this study has suggested guidelines for effective usage and management of AI by reviewing the literature and real-world uses of AI systems in healthcare organizations.



We anticipate that our research will spur additional rigorous theoretical and empirical research into the most effective use of AI systems to deliver the best possible treatment for patients and public health prevention.

When AI-driven research entails large-scale projects claiming data from individuals and communities or groups, appropriate modes of inclusion must be tested to enable social learning spanning various epistemic groups, encompassing lay public and non - academic actors.

- Due to their authority and domination over the resources and information that underlies the digital economy, global enterprises will be highly likely to govern decisions made by individual citizens, communities, and government agencies when they manage the majority of data, health analytics, and algorithms. This power differential also impacts a person who should be properly treated by their governments or, at the very least, hold their governments accountable if mistreated.
- To utilize AI in clinical practice, physicians and nurses will need a broader set of skills, including a better knowledge of mathematics, AI underlying principles, data science, health data chain of custody, collection, assimilation, and management, as well as the ethical and legal issues surrounding AI for health. In addition, such procedures (including training) will be required to integrate and analyze data from a variety of sources adequately, monitor AI tools, and recognize AI performance that is erroneous.
- When employing AI technologies, good assistance and training will ensure that healthcare personnel and physicians, for example, can avoid common issues such as automation bias. In partnership with clinicians and academics, professionals and legal, regulatory authorities may ultimately specify health workers' expertise, experiences, and abilities. The public should be included in the development of AI for health to comprehend data sharing better and use, provide feedback on culturally and socially appropriate types of AI and adequately communicate their worries and aspirations.

How will future work be done? For example, is it possible to apply ethical design explicitly to AI technology for health? What are the best ways for suppliers and programmers to handle any biases that may appear in applications?



## REFERENCES

- [1] Saxena, A., Brault, N., & Rashid, S. (2021). Big Data and Artificial Intelligence for Healthcare Applications (1st ed.). CRC Press. <https://doi.org/10.1201/9781003093770>.
- [2] Goyal, L.M., Saba, T., Rehman, A., & Larabi-Marie-Sainte, S. (Eds.). (2021). Artificial Intelligence and Internet of Things: Applications in Smart Healthcare (1st ed.). CRC Press. <https://doi.org/10.1201/9781003097204>
- [3] Sun, L., Gupta, R.K. & Sharma, A. Review and potential for artificial intelligence in healthcare. Int J Syst Assur Eng Manag (2021). <https://doi.org/10.1007/s13198-021-01221-9>
- [4] Kishor, A., Chakraborty, C. Artificial Intelligence and Internet of Things Based Healthcare 4.0 Monitoring System. Wireless Pers Commun (2021). <https://doi.org/10.1007/s11277-021-08708-5>
- [5] Johnson, M., Albizri, A. & Simsek, S. Artificial intelligence in healthcare operations to enhance treatment outcomes: a framework to predict lung cancer prognosis. Ann Oper Res (2020). <https://doi.org/10.1007/s10479-020-03872-6>



- [6] Ostherr, K. Artificial Intelligence and Medical Humanities. *J Med Humanit* (2020). <https://doi.org/10.1007/s10912-020-09636-4>
- [7] Amann, J., Blasimme, A., Vayena, E. et al. Explainability for artificial intelligence in healthcare: a multidisciplinary perspective. *BMC Med Inform Decis Mak* 20, 310 (2020). <https://doi.org/10.1186/s12911-020-01332-6>
- [8] Bohr, A., & Memarzadeh, K. (2020). The rise of artificial intelligence in healthcare applications. *Artificial Intelligence in Healthcare*, 25–60. <https://doi.org/10.1016/B978-0-12-818438-7.00002-2>
- [9] Hindocha, S., Badea, C. Moral exemplars for the virtuous machine: the clinician's role in ethical artificial intelligence for healthcare. *AI Ethics* (2021). <https://doi.org/10.1007/s43681-021-00089-6>
- [10] Winfield, A.F., Michael, K., Pitt, J., Evers, V.: Machine ethics: the design and governance of ethical ai and autonomous systems. *Proc. IEEE* 107, 509–517 (2019).



- [11] Char, D.S., Shah, N.H., Magnus, D.: Implementing machine learning in health care' addressing ethical challenges. *N. Engl. J. Med.* 378, 981–983 (2018).
- [12] Keskinbora, K.H.: Medical ethics considerations on artificial intelligence. *J. Clin. Neurosci.* 64, 277–282 (2019).
- [13] Hagendorff, T.: The ethics of AI ethics—an evaluation of guidelines. *Minds Mach.* (2019).  
<https://doi.org/10.1007/s11023-020-09517-8>
- [14] McDougall, R.J.: Computer knows best? The need for value-flexibility in medical AI. *J. Med. Ethics* 45, 156–160 (2019)
- [15] Rigby, M.J.: Ethical dimensions of using artificial intelligence in health care. *AMA J. Ethics* 21, 121–124 (2019)
- [16] Broome, D.T., Hilton, C.B. & Mehta, N. Policy Implications of Artificial Intelligence and Machine Learning in Diabetes Management. *Curr Diab Rep* 20, 5 (2020).  
<https://doi.org/10.1007/s11892-020-1287-2>



- [17] Mirbabaie, M., Stieglitz, S. & Frick, N.R.J. Artificial intelligence in disease diagnostics: A critical review and classification on the current state of research guiding future direction. *Health Technol.* 11, 693–731 (2021). <https://doi.org/10.1007/s12553-021-00555-5>
- [18] Ishii, K. Comparative legal study on privacy and personal data protection for robots equipped with artificial intelligence: looking at functional and technological aspects. *AI & Soc* 34, 509–533 (2019). <https://doi.org/10.1007/s00146-017-0758-8>
- [19] Luan, F., Gao, X., Zhao, S. et al. The Roles of Plastic Surgeons in Advancing Artificial Intelligence in Plastic Surgery. *Aesth Plast Surg* (2021). <https://doi.org/10.1007/s00266-021-02302-7>
- [20] Meskó, B., Hetényi, G. & Gyórfy, Z. Will artificial intelligence solve the human resource crisis in healthcare?. *BMC Health Serv Res* 18, 545 (2018). <https://doi.org/10.1186/s12913-018-3359-4>
- [21] Pandey, S.K., Janghel, R.R. Recent Deep Learning Techniques, Challenges and Its Applications for Medical Healthcare System: A Review. *Neural Process Lett* 50, 1907–1935 (2019). <https://doi.org/10.1007/s11063-018-09976-2>



- [22] Matsumura, N. From genetic analysis to precision medicine. *Int Canc Conf J* 10, 159 (2021). <https://doi.org/10.1007/s13691-021-00492-0>
- [23] Arnold, M.H. Teasing out Artificial Intelligence in Medicine: An Ethical Critique of Artificial Intelligence and Machine Learning in Medicine. *Bioethical Inquiry* 18, 121–139 (2021). <https://doi.org/10.1007/s11673-020-10080-1>
- [24] Wong, J., Murray Horwitz, M., Zhou, L. et al. Using Machine Learning to Identify Health Outcomes from Electronic Health Record Data. *Curr Epidemiol Rep* 5, 331–342 (2018). <https://doi.org/10.1007/s40471-018-0165-9>
- [25] Crowson, M.G., Chan, T.C.Y. Machine Learning as a Catalyst for Value-Based Health Care. *J Med Syst* 44, 139 (2020). <https://doi.org/10.1007/s10916-020-01607-5>
- [26] Alanazi, H.O., Abdullah, A.H. & Qureshi, K.N. A Critical Review for Developing Accurate and Dynamic Predictive Models Using Machine Learning Methods in Medicine and Health Care. *J Med*



Syst 41, 69 (2017). <https://doi.org/10.1007/s10916-017-0715-6>

[27] Havaei, F., Ji, X.R., MacPhee, M. et al. Identifying the most important workplace factors in predicting nurse mental health using machine learning techniques. BMC Nurs 20, 216 (2021). <https://doi.org/10.1186/s12912-021-00742-9>

[28] Srividya, M., Mohanavalli, S. & Bhalaji, N. Behavioral Modeling for Mental Health using Machine Learning Algorithms. J Med Syst 42, 88 (2018). <https://doi.org/10.1007/s10916-018-0934-5>

[29] Payrovnaziri, S. N., Chen, Z., Rengifo-Moreno, P., Miller, T., Bian, J., Chen, J. H., Liu, X., & He, Z. (2020). Explainable artificial intelligence models using real-world electronic health record data: a systematic scoping review. Journal of the American Medical Informatics Association: JAMIA, 27(7), 1173–1185. <https://doi.org/10.1093/jamia/ocaa053>

[30] Adadi A, Berrada M.. Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). IEEE Access 2018; 6: 52138–60



- [31] Tjoa E, Guan C. A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI. *IEEE Trans Neural Netw Learn Syst.* 2021 Nov;32(11):4793-4813. DOI: 10.1109/TNNLS.2020.3027314. Epub 2021 Oct 27. PMID: 33079674.
- [32] Amann, J., Blasimme, A., Vayena, E. et al. Explainability for artificial intelligence in healthcare: a multidisciplinary perspective. *BMC Med Inform Decis Mak* 20, 310 (2020). <https://doi.org/10.1186/s12911-020-01332-6>
- [33] London AJ. Artificial intelligence and black-box medical decisions: accuracy versus explainability. *Hastings Cent Rep* 2019;49:1521. Available from: <https://doi.org/10.1002/hast.973>.
- [34] Alemzadeh, H., Raman, J., Leveson, N., Kalbarczyk, Z., & Iyer, R. K. (2016). Adverse Events in Robotic Surgery: A Retrospective Study of 14 Years of FDA Data. *PloS one*, 11(4), e0151470. <https://doi.org/10.1371/journal.pone.0151470>
- [35] Sujan Sarker, Lafifa Jamal, Syeda Faiza Ahmed, Niloy Irtisam, Robotics and artificial intelligence in healthcare during COVID-



19 pandemic: A systematic review, *Robotics and Autonomous Systems*, Volume 146, 2021,

[36] Liyanage, H., Liaw, S. T., Jonnagaddala, J., Schreiber, R., Kuziemy, C., Terry, A. L., & de Lusignan, S. (2019). Artificial Intelligence in Primary Health Care: Perceptions, Issues, and Challenges. *Yearbook of medical informatics*, 28(1), 41–46. <https://doi.org/10.1055/s-0039-1677901>

[37] Liu, N., Shapira, P., & Yue, X. (2021). Tracking developments in artificial intelligence research: constructing and applying a new search strategy. *Scientometrics*, 126(4), 3153–3192. <https://doi.org/10.1007/s11192-021-03868-4>

[38] Ntoutsis, E., Fafalios, P., Gadiraju, U, et al. Bias in data-driven artificial intelligence systems—An introductory survey. *WIREs Data Mining Knowl Discov.* 2020; 10:e1356. <https://doi.org/10.1002/widm.1356>

[39] Vaid, S., Kalantar, R. & Bhandari, M. Deep learning COVID-19 detection bias: accuracy through artificial intelligence. *International Orthopaedics (SICOT)* 44, 1539–1542 (2020). <https://doi.org/10.1007/s00264-020-04609-7>



- [40] Kaissis, G.A., Makowski, M.R., Rückert, D. et al. Secure, privacy-preserving and federated machine learning in medical imaging. *Nat Mach Intell* 2, 305–311 (2020). <https://doi.org/10.1038/s42256-020-0186-1>
- [41] Sri S., Ferry Fadzlul R., Muhammad N., Muhammad R., Kresna F., Rusni M., Artificial intelligence in healthcare: opportunities and risk for future, *Gaceta Sanitaria*, Volume 35, Supplement 1, 2021, Pages S67-S70, ISSN 0213-9111.
- [42] Manne, Ravi & Kantheti, Sneha. (2021). Application of Artificial Intelligence in Healthcare: Chances and Challenges. *Current Journal of Applied Science and Technology*. 40. 78-89. 10.9734/CJAST/2021/v40i631320.
- [43] McNally R, Alborz A. Developing methods for systematic reviewing in health services delivery and organization: an example from a review of access to health care for people with learning disabilities. Part 1. Identifying the literature. *Health Info Libr J* 2004;21:182–92. 10.1111/j.1471-1842.2004.00512.x



- [44] Ye K., Liu Y., Xu G., Xu CZ. (2018) Fault Injection and Detection for Artificial Intelligence Applications in Container-Based Clouds. In: Luo M., Zhang LJ. (eds) Cloud Computing – CLOUD 2018. CLOUD 2018. Lecture Notes in Computer Science, vol 10967. Springer, Cham. [https://doi.org/10.1007/978-3-319-94295-7\\_8](https://doi.org/10.1007/978-3-319-94295-7_8)
- [45] Molnár-Gábor F. (2020) Artificial Intelligence in Healthcare: Doctors, Patients and Liabilities. In: Wischmeyer T., Rademacher T. (eds) Regulating Artificial Intelligence. Springer, Cham. [https://doi.org/10.1007/978-3-030-32361-5\\_15](https://doi.org/10.1007/978-3-030-32361-5_15)
- [46] Zielke T. (2020) Is Artificial Intelligence Ready for Standardization?. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham. [https://doi.org/10.1007/978-3-030-56441-4\\_19](https://doi.org/10.1007/978-3-030-56441-4_19) Russia, Evgeniy. (2020).
- [47] Standardization of Artificial Intelligence for the Development and Use of Intelligent Systems. Advances in Wireless



Communications and Networks. 6. 1.  
10.11648/j.awcn.20200601.11.

[48] Coghlan, S. Robots and the Possibility of Humanistic Care. *Int J of Soc Robotics* (2021). <https://doi.org/10.1007/s12369-021-00804-7>

[49] Martani, A., Geneviève, L.D., Poppe, C. et al. Digital pills: a scoping review of the empirical literature and analysis of the ethical aspects. *BMC Med Ethics* 21, 3 (2020). <https://doi.org/10.1186/s12910-019-0443-1>

[50] Neri, E., Coppola, F., Miele, V. et al. Artificial intelligence: Who is responsible for the diagnosis?. *Radiol med* 125, 517–521 (2020). <https://doi.org/10.1007/s11547-020-01135-9>



LaSalle  
Universidad  
Perú