



Instituto Politécnico Nacional
Unidad Profesional Interdisciplinaria de
Ingeniería campus Zacatecas

Área de ubicación para el desarrollo del
trabajo

Ingeniería en Sistemas Computacionales

Línea de investigación

Desarrollo de sistemas.

Título del proyecto de Trabajo Terminal

Software para generación de código CNC para
corte de plasma en metales.

Presenta(n):

Pedro Alejandro Nunez Perez

Director:

M. en C. Eleazar Pacheco Reyes

Asesores:

M. en C. Erika Paloma Sánchez Femat



Zacatecas, Zacatecas a 6 de Enero de 2025

Firmas.

En esta sección se mostrarán los nombres y las firmas de los alumnos responsables del desarrollo del proyecto de Trabajo Terminal.

C. Alumno 1.

C. Alumno 2.

Autorización.

Por medio del presente autorizo la impresión y distribución del presente reporte final de proyecto de Trabajo Terminal, toda vez que lo he leído, comprendido en su totalidad, y estoy de acuerdo con su contenido.

Atentamente;

Grado Nombre y firma del director del proyecto de TT

DIRECTOR

Grado Nombre y firma del asesor 1.

ASESOR 1

Grado Nombre y firma del asesor 2.

ASESOR 2

Índices

Índice de contenido

Firmas.	i
Autorización.	i
Resumen.	1
Abstract.....	1
Definición del problema.	1
Contexto y antecedentes generales del problema.	2
Situación problemática o problema de investigación.	2
Estado del arte.	2
Descripción del proyecto.	4
Objetivo general del proyecto.....	5
Objetivos particulares del proyecto.	5
Justificación.	5
Hipótesis.	Error! Bookmark not defined.
Marco teórico.....	6
Marco Metodológico.	7
Análisis y Discusión de los Resultados	10
Gestión del proyecto	11
1. Plan del proyecto.	11
2. Manejo de desviaciones en la ejecución del plan.	12
3. Plan de los riesgos del proyecto.	12
Desarrollo del proyecto	Error! Bookmark not defined.
1. Resumen del análisis del sistema.....	13
2. Diseño del sistema.	15
a. Arquitectura del sistema.	15
b. Matriz de trazabilidad.....	15
c. Diseño de la base de datos (cuando aplique).....	Error! Bookmark not defined.

d. Manejo de archivos (cuando aplique).....	17
e. Construcción.....	17
f. Seguimiento al plan de pruebas.	25
g. Entrega o liberación.....	25
Conclusiones y Recomendaciones.....	26
Fuentes de consulta.....	28
Apéndices	29

Índice de tablas

Índice de figuras

Resumen.

El objetivo del proyecto es simplificar la generación de código G, teniendo como propósito la creación de un sistema el cual con el análisis de archivos DXF y mediante una inteligencia artificial genere el código G, teniendo en cuenta diferentes aspectos como la variabilidad de las figuras y la complejidad de la trayectoria del corte; todo esto con la finalidad de poder simplificar este proceso en la generación de piezas. La dificultad que se tiene es que actualmente la generación de este tipo de código es un proceso complejo que requiere un conocimiento amplio de la geometría de las piezas y los parámetros de corte.

Palabras clave: Inteligencia artificial, Código G, Análisis de imágenes, Automatización, Simplificación del proceso.

Abstract.

Deberá escribirse el resumen en el idioma inglés.

El

Keys Word: Deberá escribirse las palabras clave en el idioma inglés.

Definición del problema.

En [2] se dice que el código G es un lenguaje de programación poco amigable, por lo que en los últimos años las personas han preferido aprender lenguajes de un nivel mayor; sin embargo, esto no quiere decir que este lenguaje esté quedando obsoleto, todo lo contrario, sigue siendo un lenguaje muy utilizado en específico por la industria manufacturera, por lo que la generación de un sistema el cual pueda automatizar este proceso sería crucial debido a la importancia que tiene este código en la industria actualmente; además, como se menciona en [2], es necesario descubrir la automatización en CNC, ya que la mayoría del área de manufactura utiliza las máquinas CNC para la generación de piezas requeridas. Y esto podría ayudar a simplificar este proceso que puede llegar a consumir muchos recursos dentro de la industria manufacturera.

Contexto y antecedentes generales del problema.

Actualmente, no existe ningún proyecto documentado para la automatización de este proceso; sin embargo, hay proyectos con los cuales se puede simplificar un poco el proceso. Uno de ellos es el proyecto mencionado en [1] que consiste en una librería para Python que transforma el código generado en este lenguaje de programación y lo traduce a código G. Además, también se habla a grandes rasgos de la necesidad de simplificar este proceso, ya que, aunque este lenguaje tiene una amplia utilidad en la industria, entre los desarrolladores está surgiendo la tendencia de aprender los lenguajes de alto nivel por la dificultad que representa la generación de código G.

Situación problemática o problema de investigación.

Aunque el proyecto anterior se muestra como una manera sencilla de programar el código G, sigue sin satisfacer la necesidad de tener un sistema que lo automatice. La presente propuesta busca simplificar aún más este proceso mediante el desarrollo de un sistema que mediante el análisis de archivos DXF (Design Exchange Format); esto debido a que estos dos formatos son los que más comúnmente se relacionan con los programas CAD (diseño asistido por computadora) y con uso de una inteligencia artificial, logre generar el código G para la figura deseada y que además logre identificar una secuencia de corte para que la pieza que se desea cortar no sufra movimientos ni errores, esto con el propósito de poder simplificar y automatizar este proceso y de esta forma ahorrar tiempos en la industria en la que es necesitada esta tecnología, en [3] se menciona que es necesario descubrir la automatización de este proceso para la industria, ya que la generación de este código tiene una complejidad muy alta, lo que está provocando que este lenguaje tenga una complejidad de aprendizaje muy alta.

Estado del arte.

La inteligencia artificial ha transformado innumerables áreas de la actividad humana, pero todavía existen ámbitos donde su potencial no ha sido plenamente explorado, como la generación automatizada de código G. Aunque las herramientas existentes han facilitado el desarrollo de este lenguaje de control industrial, la automatización completa sigue siendo un

reto significativo. Automatizar este proceso no solo reduciría el tiempo necesario para diseñar piezas industriales, sino que también haría el código G más accesible para personas sin experiencia técnica. Este proyecto propone el uso de modelos basados en Transformers, aprovechando los avances recientes en inteligencia artificial, para abordar este desafío de manera innovadora.

Como se menciona en [10], los Transformers destacan por su capacidad para trabajar con datos secuenciales etiquetados o no etiquetados, generando contenido coherente y adaptado a necesidades específicas.

Los avances recientes en inteligencia artificial han permitido el desarrollo de diversos métodos de aprendizaje, entre los cuales destacan:

Tabla 1 Métodos de aprendizaje

CARACTERÍSTICA	APRENDIZAJE SUPERVISADO	APRENDIZAJE NO SUPERVISADO	TRANSFORMERS
TIPO DE DATOS	Datos etiquetados	Datos no etiquetados	Secuencias etiquetadas/no etiquetadas
OBJETIVO	Aprender una función que mapee entradas a salidas	Descubrir patrones en los datos	Generar secuencias estructuradas
EJEMPLOS	Clasificación, regresión	Agrupamiento, reducción de dimensionalidad, detección de anomalías	Traducción automática, generación de código G
REQUERIMIENTOS DE DATOS	Se requieren datos etiquetados, costosos de obtener	Se requieren datos no etiquetados	Secuencias que capturen dependencias definidas
VENTAJAS	Eficiente para tareas de clasificación y regresión	Útil para descubrir patrones complejos	Captura relaciones de largo alcance

DESVENTAJAS	Dificultad para obtener datos etiquetados	Interpretación de patrones descubiertos	Requiere alto costo computacional y grandes datos
--------------------	---	---	---

Considerando las necesidades de este proyecto, se emplearán Transformers debido a su capacidad para procesar y generar contenido estructurado y coherente. Inspirados por las ideas presentadas en [10], los Transformers ofrecen una solución avanzada para abordar la complejidad de las secuencias, como el código G, gracias a su capacidad de capturar relaciones a largo plazo dentro de los datos.

El objetivo es entrenar un modelo generador basado en Transformers que sea capaz de producir código G que cumpla con los requisitos del diseño industrial. Este modelo aprovecha el mecanismo de atención para identificar dependencias clave en el lenguaje de control numérico, mientras que sus características de generación garantizan que las secuencias creadas sean válidas y funcionales.

Esta metodología elimina la necesidad de especificar explícitamente cada paso del proceso. En su lugar, el modelo aprende de manera autónoma a desarrollar código G, reduciendo significativamente los tiempos de diseño y optimizando su integración en flujos de trabajo industriales.

En resumen, este proyecto busca demostrar cómo los Transformers pueden ser una herramienta poderosa para automatizar la generación de código G. Esto no solo marca un avance significativo en la aplicación de inteligencia artificial a problemas prácticos en la industria, sino que también destaca las capacidades únicas de los Transformers, como se describe en el artículo mencionado, para transformar tareas tradicionales en soluciones más eficientes y modernas.

Descripción del proyecto.

Este proyecto se enfoca en el diseño e implementación de un sistema, el cual, a partir del

análisis de archivos DXF (las cuales tendrán que ser ingresadas por el usuario), el sistema generará un archivo de texto plano con el código g, que será producido por una inteligencia artificial. Este archivo de texto plano que genere el sistema tendrá que ser un archivo TXT (archivo de texto plano) debido a las necesidades que tiene la máquina de corte por plasma que nos proporcionará el cliente. El sistema contará con una interfaz de usuario donde se podrán cargar los archivos del tipo antes mencionados, en donde podrá modificar ciertos aspectos del corte como la velocidad del corte y la altura de la antorcha de corte. Durante el proceso del análisis basado en recompensas, el sistema tendrá que identificar la parte del material que quedará inutilizada para comenzar el corte en esa zona, debido a que al iniciar el corte la máquina deja un error que es indeseable para los resultados esperados.

Objetivo general del proyecto.

Obtener código G a partir del análisis de archivos DXF mediante un proceso de inteligencia artificial.

Objetivos particulares del proyecto.

- Obtener parámetros de corte de una interfaz de usuario.
- Establecer un método de carga de archivos DXF.
- Detectar los bordes de figuras mediante un proceso de análisis archivos DXF.
- Establecer un método basado en aprendizaje por refuerzo que sea capaz de generar código G preciso y funcional para cortar las piezas a partir de las características geométricas reconocidas.
- Establecer un método de salida para extraer el código G generado.

Justificación.

La industria de manufactura, especialmente aquella que utiliza máquinas CNC (Control Numérico Computarizado), se enfrenta a desafíos significativos en la generación de código G, un lenguaje de programación esencial para el control preciso de herramientas de corte,

fresado y otros procesos. Este proyecto surge como respuesta a la necesidad de simplificar y agilizar este proceso mediante la aplicación de inteligencia artificial, específicamente a través de un enfoque basado en aprendizaje por refuerzo. Esto debido a la complejidad que generar código G genera, ya que es conocido por ser poco amigable y requerir un conocimiento profundo de la geometría de las piezas; a esto se le agrega que en años recientes ha crecido el mercado de las maquina CNC como se menciona en [7].

Con este proyecto no solo se busca automatizar la generación de código G, sino también tener un impacto en la educación debido a que los precios que llegar a alcanzar los softwares CAM (Manufactura Asistida por Computadora) son demasiado altos según [8], este sistema podría ser una alternativa para que las instituciones educativas que no tengan un gran poder adquisitivo puedan generar este código, ya que este software será de gran ayuda para los profesores de esta área, ya que tan solo actualmente al menos ingresan 80 alumnos a nuestra unidad académica en la carrera de ingeniería mecatrónica, en la cual desde los primeros semestres se le imparten materia relacionadas al tema.

Marco teórico.

En este apartado se abordan los conceptos básicos necesarios de conocer para una mejor comprensión del proyecto, ya que son los que sustentan el presente trabajo.

Inteligencia Artificial

Como producto esperado del presente trabajo, se espera un modelo de IA (Inteligencia Artificial) esto se define como “la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano. [10]”; el cual a partir del procesamiento de archivos DXF se obtenga código G eficiente y funcional extraíble en un archivo TXT para su utilización en máquinas de corte por plasma.

Aprendizaje por refuerzo

El modelo de IA será entrenado por un método de aprendizaje por refuerzo el cual es uno de los enfoques más usados dentro del aprendizaje, lo que lo diferencia de los demás métodos de aprendizaje es que “el agente obtiene experiencia útil acerca de los estados, acciones, transiciones y recompensas de manera activa para poder actuar de manera óptima [6]” y que la evaluación del modelo ocurre de forma concurrente con el aprendizaje.

Automatización

La automatización de procesos de fabricación ha avanzado ampliamente, ya que actualmente la mayoría de las empresas donde se requiere la fabricación de un producto llevaron un proceso de automatización, esto para llevar su producto hacia una estandarización de los procesos ya que pagar a un especialista en una acción en específico es caro, en este caso, la generación de código G implica un gran conocimiento de este lenguaje, y de la geometría de las piezas [4]; esta automatización generaría la estandarización de generación de código G, además de mejorar la eficiencia y reducir costos [3]

Interfaz de usuario

Una implementación de una interfaz de usuario o interfaz gráfica. En este proyecto es de suma importancia debido a que será la forma en la que el usuario se podrá comunicar con el modelo de IA, esto mediante la carga de archivos y parámetros de corte, también de forma útil el usuario podrá decidir, mediante la visualización de una simulación de lo generado por el modelo, si este código es útil. Esto con la finalidad de generar una experiencia más amigable para el usuario.

Marco Metodológico.

Las metodologías ágiles son aquellas las cuales, según Kumar et. al. [1] están basadas en desarrollo incremental e iterativo. Estas tienen cuatro principales características, las cuales son: planeación adaptativa, iterativa, desarrollo evolutivo y rápida adaptabilidad al cambio, lo que lo hace más adecuado para proyectos pequeños.

El desarrollo agile divide las tareas en pequeños incrementos con muy poca planeación llamados iteraciones.

Como se menciona en [5], scrum es un framework ágil de desarrollo de software iterativo e incremental para el manejo de desarrollo de productos, esto siendo de gran ventaja para el desarrollo de este proyecto debido a que este nos permite hacer modificaciones de una forma sencilla en caso de que se requiera agregar o eliminar algún proceso. Este framework de desarrollo tiene diferentes partes en las cuales lo podemos dividir, las cuales son: épicas, tareas y backlog.

Además, también existen diferentes ceremonias o eventos, los cuales son: sprint planning,

sprint review, sprint, sprint retrospective y daily scrum meeting. Cuenta asimismo con los roles de product owner, scrum máster y equipo de desarrollo.

- Sprint planning: Los sprint planning son las reuniones en las que se planean las iteraciones.
 - Tienen un promedio de ocho horas para un mes de sprint
 - Es llevada a cabo por el Scrum máster
- Sprint Review: Es una reunión con un tiempo de 2 a 4 horas de duración por cada sprint:
 - Se realiza al finalizar el sprint y es donde el equipo presenta las tareas que se planearon durante el sprint planning
 - Es llevada a cabo por el Scrum máster
- Sprint retrospective: Se discute que fue bien durante el sprint y que fue mal
 - Asiste el product owner así como también el equipo de desarrollo
 - Es dirigida por el Scrum máster
- Dayli scrum meeting: Es una reunión diaria de 15 a 20 minutos en donde se reúne el equipo de trabajo y el Scrum master y se menciona que se hizo el día anterior de trabajo y que es lo que se hará durante ese día
- Sprint: Es el periodo de tiempo en el que se realizaran las iteraciones planeadas durante el sprint planning

Esta metodología fue elegida debido a la adaptabilidad al cambio que tiene. Además de aún no tener muy bien segmentadas las tareas por hacer, nos permite comenzar con el desarrollo y, en caso de que exista un requerimiento que no se tenía en cuenta desde el inicio del proyecto, se puede agregar en alguna iteración.

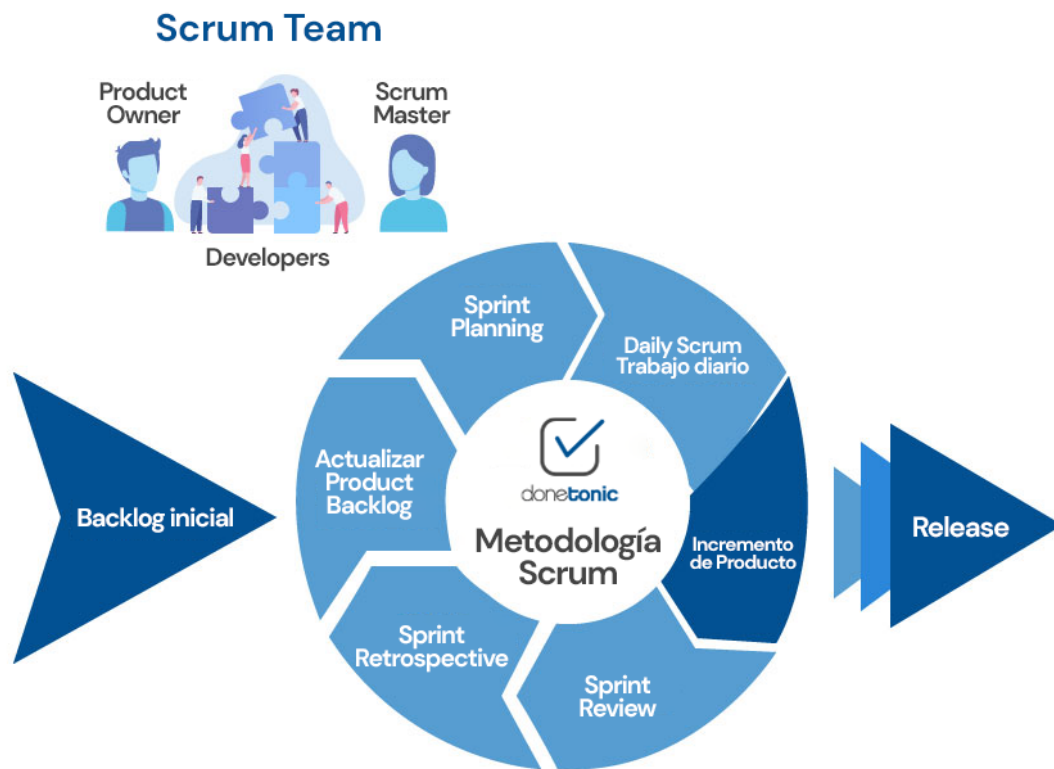


Figura 1 SCRUM

La forma en la que adaptaremos las características antes mencionadas será de la siguiente manera; estarán fungiendo como equipo Pedro Alejandro Nunez Perez, la M. en C. Erika Paloma Sánchez Femat fungio como Scrum máster y el M. en C. Eleazar Pacheco Reyes como product owner, que es el intermediario con el cliente que es el Dr. Miguel Fernando Delgado Pamanes.

El backlog de nuestro proyecto (debido a que aún no se tienen requerimientos muy detallados por parte del cliente) se presenta como un backlog de una manera muy general, teniendo en cuenta que los objetivos particulares serán tomados como las épicas de nuestro backlog; estas épicas tendrán que ser divididas en tareas más pequeñas, luciendo de la siguiente forma:

▼ ⚡ Darle funcionalidad 4		
Aa Nombre de la tarea	🕒 Tiempo	🕒 Estimaciones
Implementación de la funcionalidad de carga de archivos	4	2
📄 Implementación de la funcionalidad de descarga de código G	3	2
📄 Desarrollo de la lógica para ajustar parámetros de corte	4	5
📄 Documentación	2	3
+ Nuevo ítem		
▼ 🛠 Hacer Interfaces 4 ... +		
Aa Nombre de la tarea	🕒 Tiempo	🕒 Estimaciones
📄 Pruebas de usabilidad	3	3
📄 Configurar el entorno de desarrollo	10	3
📄 Codificación de los prototipos	32	8
📄 Documentación	2	3

Figura 2 Backlog p.1

🧠 Modelar IA 6		
Aa Nombre de la tarea	🕒 Tiempo	🕒 Estimaciones
📄 Preparación de los datos	8	3
📄 Capacitación	40	8
📄 Entrenar		8
📄 Probar		8
📄 Implementar		8
📄 Documentacion	2	3
Nuevo ítem		
⚠ Prototipos 5 ... +		
Aa Nombre de la tarea	🕒 Tiempo	🕒 Estimaciones
📄 Creación de bocetos iniciales de la interfaz de usuario.	5	1
📄 Desarrollo de prototipos de alta fidelidad utilizando herramientas de diseño.	20	8
Revisión y retroalimentación de los prototipos con los stakeholders.	2	5
Ajustes basados en la retroalimentación	8	1
📄 Documentacion	5	3

Figura 3 Backlog p.2

Para las daily's al proyecto, tener solo un desarrollador es suficiente; no es tan necesaria la realización de estas reuniones.

Como apoyo para poder conllevar de una manera más entendible esta metodología, se utilizo el software de notion, el cual facilita herramientas para su entendimiento.

Análisis y Discusión de los Resultados

Gestión del proyecto

1. Plan del proyecto.

Para la ejecución del plan de proyecto se tuvieron que realizar algunos ajustes conforme a las fechas que se nos dieron para la entrega de los documentos, estos ajustes se pueden observar en la Figura 4.

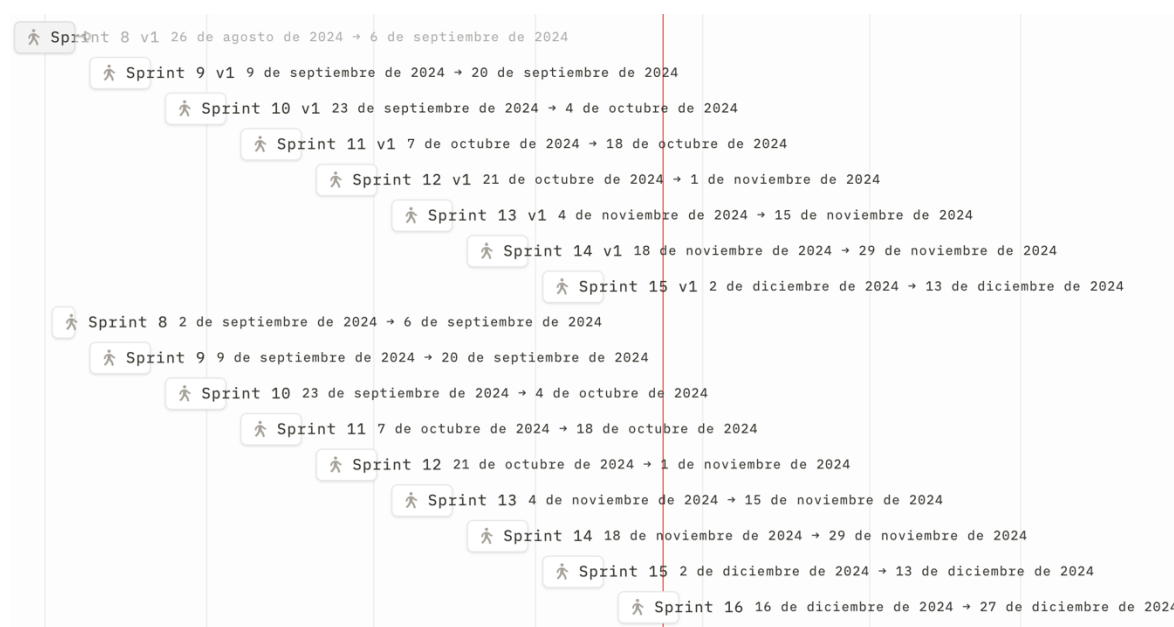


Figura 4 Plan de proyecto

A continuación se muestran los trabajos realizados durante los diferentes Sprints que fueron planeados durante los Sprint Planning.

Aa Sprint name	📌 Is Current ...	📅 Dates	📋 Tasks
🚩 Sprint 16	<input checked="" type="checkbox"/>	16 de diciembre de 2024 → 27 de diciembre de 2024	<div>📄 Entrenar</div> <div>📄 Probar</div> <div>📄 Implementar</div> <div>📄 Documentacion</div>
🚩 Sprint 15	<input type="checkbox"/>	2 de diciembre de 2024 → 13 de diciembre de 2024	<div>📄 Documentación</div> <div>📄 Preparación de los datos</div> <div>📄 Entrenar</div> <div>📄 Probar</div>
🚩 Sprint 14	<input type="checkbox"/>	18 de noviembre de 2024 → 29 de noviembre de 2024	<div>📄 Documentación</div> <div>📄 Preparación de los datos</div> <div>📄 Entrenar</div>
🚩 Sprint 13	<input type="checkbox"/>	4 de noviembre de 2024 → 15 de noviembre de 2024	<div>📄 Desarrollo de simulador de código G</div> <div>📄 Preparación de los datos</div>

Figura 5 Ejecutado p.1

🚩 Sprint 12	<input type="checkbox"/>	21 de octubre de 2024 → 1 de noviembre de 2024	📄 Desarrollo de simulador de código G
🚩 Sprint 11	<input type="checkbox"/>	7 de octubre de 2024 → 18 de octubre de 2024	📄 Desarrollo de simulador de código G
🚩 Sprint 10	<input type="checkbox"/>	23 de septiembre de 2024 → 4 de octubre de 2024	📄 Pruebas de usabilidad 📄 Codificación de los prototipos 📄 Desarrollo de la lógica para ajustar parámetros de corte 📄 Implementación de la funcionalidad de carga de archivos 📄 Implementación de la funcionalidad de descarga de código G 📄 Desarrollo de simulador de código G
🚩 Sprint 9	<input type="checkbox"/>	9 de septiembre de 2024 → 20 de septiembre de 2024	📄 Desarrollo de simulador de código G 📄 Documentación
🚩 Sprint 8	<input type="checkbox"/>	2 de septiembre de 2024 → 6 de septiembre de 2024	📄 Capacitación

Figura 6 Ejecutado p.2

Algo que llevo más tiempo del que se tenía pensado y que de hecho no se pudo llevar a cabo por los “bugs” o problemas de desarrollo fue el simulador de código G, debido a esto se implementó que en el momento en el que se quisiera utilizar el simulador, el programa mandara al usuario a una página web gratuita para poder observar el código G que se generó, esto para solucionar la problemática que se tuvo para implementar esta función.

2. Manejo de desviaciones en la ejecución del plan.

Durante la segunda etapa del desarrollo del proyecto surgieron 2 desviaciones de lo que se tenía planeado según los prototipos que se hicieron en Trabajo terminal I, los cuales son:

- Simulador del código G: Este se cambió debido a los problemas que estaba generando su codificación, ya que, al momento de aplicar las pruebas, comenzaron a surgir varios “bugs”, por lo que se optó por sustituirlo con un hipervínculo hacia un simulador web totalmente gratuito.
- Visualización de los archivos DXF: Esta función fue removida debido al tiempo que conllevaría poder realizarla, por lo que se optó que en la pantalla principal solo se mostraran los campos que se requieren para cargar los datos de las maquinas.

3. Plan de los riesgos del proyecto.

Para esta segunda etapa de desarrollo del proyecto se detonaron los riesgos que aparecen en Tabla 2, además se muestra las acciones que se tomaron para poder mitigar estos riesgos.

Tabla 2 Riesgos detonados

ID	Descripción	Causa del riesgo	Estrategia de mitigación
R-005	Disponibilidad de datos insuficiente para el entrenamiento del modelo de IA.	Falta de acceso a fuentes de datos relevantes o baja calidad de los datos disponibles.	Se trabajo y entreno la GAN con los datos proporcionados por el cliente.
R-006	Complejidad técnica del desarrollo del modelo de IA.	Desafíos técnicos como la selección de algoritmos adecuados y la gestión de la complejidad del modelo.	Se realizaron más reuniones con la asesora del proyecto para poder comprender mejor el problema, ademas de que se exploraron diversos casos.
R-011	Sobrecarga de trabajo	Subestimar el esfuerzo requerido para completar las tareas.	Se redistribuyeron las tareas a mas Sprints

1. Resumen del análisis del sistema.

Tabla 3 Requerimientos Funcionales

Nombre	Nombre corto	Descripción
Configuración de la máquina.	RF001	Permite al usuario configurar los parámetros de la máquina de corte, incluyendo velocidad de avance, amperaje, diámetro del plasma y dirección del corte, métricas inglesa y Métrica.
Navegación de lienzo	RF002	Permite al usuario moverse por el lienzo para visualizar diferentes partes de la pieza o diseño.
Generar código G	RF003	Permite al usuario generar el código G necesario para el corte basado en las

		configuraciones y diseño seleccionados.
Visualización de simulación del código G	RF004	Permite al usuario verificar la precisión y calidad del corte antes de la ejecución.
Navegador de archivos para carga o arrastre de archivos	RF005	Proporciona una interfaz para que el usuario cargue archivos de diseño, con la opción de arrastrar y soltar.
Nombre	Nombre corto	Descripción

Tabla 4 Requerimientos no funcionales

Nombre	Nombre corto	Descripción
Predominante blanco y negro	RNF001	El diseño de la interfaz debe seguir una estética predominantemente en blanco y negro.
Evitar sobreescritura de en las demás piezas	RNF002	El sistema debe evitar la sobreescritura accidental de archivos o datos de piezas existentes.
Descarga de archivo de texto con el código G.	RNF003	Facilita el uso del código G con otros sistemas o software relacionados.
Nombre	Nombre corto	Descripción

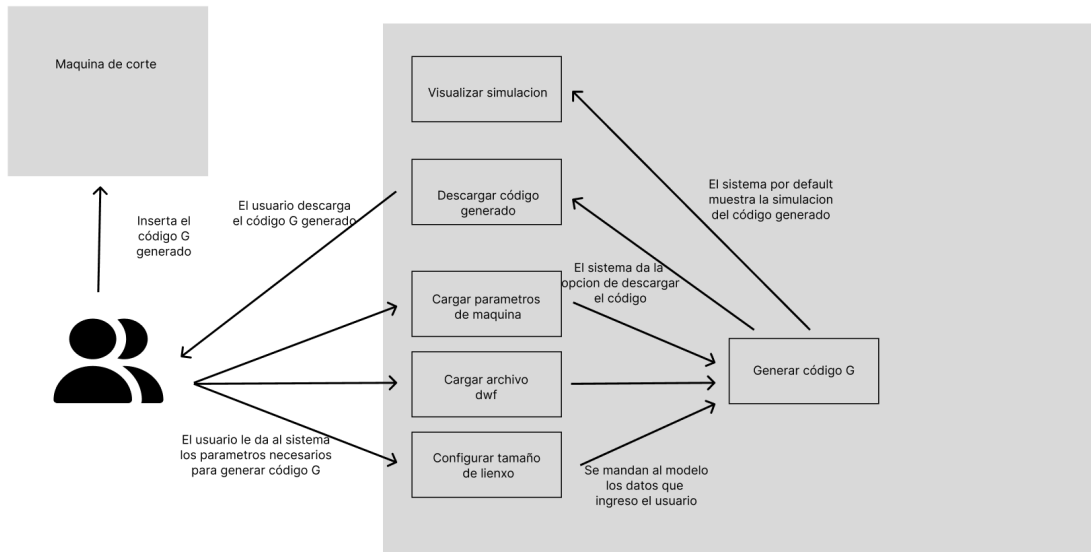


Figura 7 Conceptual de requerimientos

2. Diseño del sistema.

a. Arquitectura del sistema.

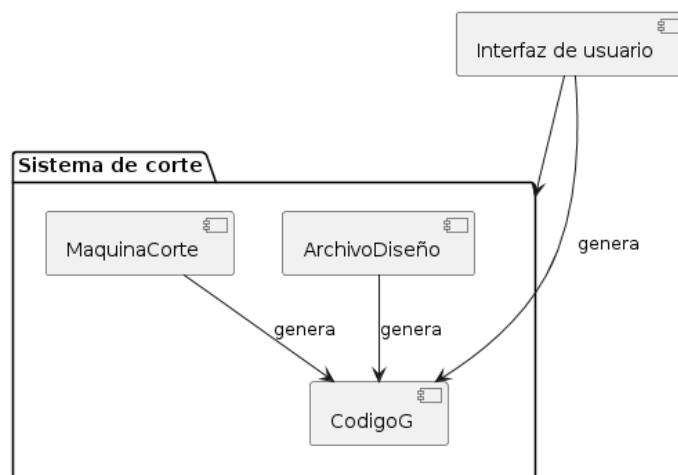


Figura 8 Arquitectura


b. Matriz de trazabilidad

Objetivo	Requerimiento	Diseño	Componente	Prueba
01 - Obtener código G a partir del análisis de imágenes y/o archivos DFX o DWG mediante un proceso de inteligencia artificial.	RF003 - Generar código G	ACT004	Código G	EP 001
	RF005 - Navegador de archivos para carga o arrastre de archivos	ACT005	ArchivoDiseño	EP004
	RNF003 - Descarga de archivo de texto con el código G	ACT006	Código G	EP 002
02 - Obtener parámetros de corte de una interfaz de usuario.	RF001 - Configuración de la máquina	ACT001	MaquinaCorte	EP008
	RF002 - Navegación de lienzo	ACT003	Interfaz de usuario	EP005
03 - Establecer un método de carga de archivos DXF.	RF005 - Navegador de archivos para carga o arrastre de archivos	ACT005	ArchivoDiseño	EP004
04 - Establecer un método basado en aprendizaje por refuerzo que sea capaz de generar código G preciso y funcional para cortar las piezas a partir de las características geométricas reconocidas.	RF004 - Visualización de simulación del código G	ACT002	Código G	EP003
				EP007
	RF003 - Generar código G			EP001
05 - Establecer un método de salida para extraer el código G.	RNF003 -	ACT006	Código G	EP002
				EP006
Objetivo	Requerimiento	Diseño	Componente	Prueba

c. Manejo de archivos.

Para el manejo de archivos que se debe tener para el proyecto el archivo tendrá que estar en el formato DXF para que el sistema permita la carga de los mismos.

d. Construcción.



The screenshot displays a web application interface with a light gray background and a dark gray header. The interface is organized into several sections:

- Parámetros de la Máquina:** Located on the left, it includes a dropdown menu labeled "Seleccionar Máquina", a text input field for "Velocidad de avance (MM/MIN)", and another text input field for "Amperaje".
- Más Parámetros:** In the center, it features a text input for "Diámetro (MM)", a dropdown for "Dirección", a "Guardar Máquina" button, and a section for "Dimensiones del lienzo" with inputs for "Altura" and "Ancho".
- Agregar Nueva Máquina:** On the right, it contains a text input for "Nombre de la Máquina", an "Agregar Nueva Máquina" button, a "Generar código G" button, and a "Cargar archivo" button with a file icon.
- Footer:** At the bottom right, there is a red prohibition icon and the text "Archivo: No se ha cargado ningun archivo".

Figura 9 Pantalla principal

En Figura 9 observamos la pantalla que se le muestra al usuario al iniciar la ejecución de la aplicación, se pueden observar los campos de texto para los parámetros de la máquina, así como también los parámetros de los lados de la pieza.

En Figura 10 observamos el caso en el que el usuario intenta guardar una máquina, faltándole un parámetro de corte, debido a esta situación el sistema lanza una alerta donde se le indica al usuario que ningún campo debe de estar vacío.

Más Parámetros

Parámetros de la Máquina

Seleccionar Máquina

Maquina

Velocidad de avance (MM/MIN)

Amperaje

12

Diámetro (MM)

5

Dirección

Centro

Guardar Máquina

Dimensiones del lienzo

Altura

Ancho

Agregar Nueva Máquina

Nombre de la Máquina

Agregar Nueva Máquina

Generar código G

Cargar archivo

Archivo: No se ha cargado ningun archivo

Ningún campo puede estar vacío

Figura 10 Validador en pantalla principal

En Figura 11 podemos observar que, aunque se han cargado la mayoría de los datos, el sistema no habilita el botón para generar código G, este se podrá habilitar hasta que el usuario, haya agregado todos los campos requeridos.

Más Parámetros

Parámetros de la Máquina

Seleccionar Máquina

Maquina

Velocidad de avance (MM/MIN)

600

Amperaje

12

Diámetro (MM)

5

Dirección

Centro

Guardar Máquina

Dimensiones del lienzo

Altura

1000

Ancho

1000

Agregar Nueva Máquina

Nombre de la Máquina

Agregar Nueva Máquina

Generar código G

Cargar archivo

Archivo: No se ha cargado ningun archivo

Figura 11 Datos faltantes para la generación

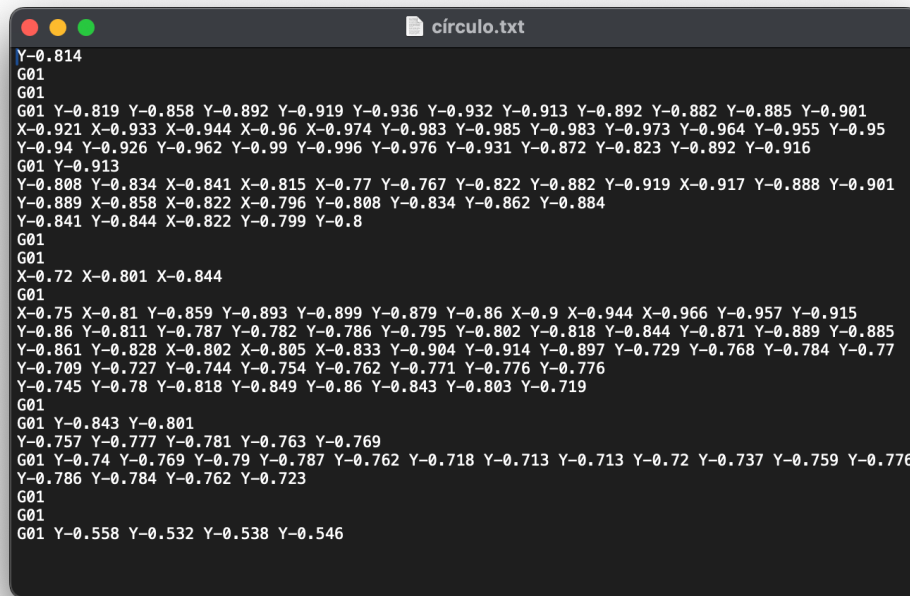
The screenshot shows a web application interface with a light gray background and a dark gray header. The interface is organized into several sections:

- Parámetros de la Máquina:** A section on the left containing a dropdown menu labeled "Seleccionar Máquina" with "Maquina" selected, and two input fields: "Velocidad de avance (MM/MIN)" with the value "600" and "Amperaje" with the value "12".
- Más Parámetros:** A central section with three sub-sections:
 - Diámetro (MM):** An input field with the value "5".
 - Dirección:** A dropdown menu with "Centro" selected.
 - Dimensiones del lienzo:** Two input fields, "Altura" with "1000" and "Ancho" with "1000".
- Guardar Máquina:** A black button with white text located below the "Más Parámetros" section.
- Agregar Nueva Máquina:** A section on the right containing a text input field for "Nombre de la Máquina", a black button labeled "Agregar Nueva Máquina", and a green button labeled "Generar código G".
- Cargar archivo:** A black button with a file icon and the text "Cargar archivo".
- Archivo:** A green checkmark followed by the text "Archivo: círculo.dxf".

Figura 12 Pantalla principal con datos cargados

En Figura 12 observamos que el sistema ya habilito el uso del botón para generar código G, esto es ya que el usuario ya cargo los campos requeridos, así como también el archivo .dxf del que se extraerán las coordenadas.

En la Figura 13 se observa la pantalla a la que nos lleva el sistema al presionar el botón para generar código G, se muestran las dos opciones disponibles que son “Descargarlo” y “Simular”.



```
Y-0.814
G01
G01
G01 Y-0.819 Y-0.858 Y-0.892 Y-0.919 Y-0.936 Y-0.932 Y-0.913 Y-0.892 Y-0.882 Y-0.885 Y-0.901
X-0.921 X-0.933 X-0.944 X-0.96 X-0.974 Y-0.983 Y-0.985 Y-0.983 Y-0.973 Y-0.964 Y-0.955 Y-0.95
Y-0.94 Y-0.926 Y-0.962 Y-0.99 Y-0.996 Y-0.976 Y-0.931 Y-0.872 Y-0.823 Y-0.892 Y-0.916
G01 Y-0.913
Y-0.808 Y-0.834 X-0.841 X-0.815 X-0.77 Y-0.767 Y-0.822 Y-0.882 Y-0.919 X-0.917 Y-0.888 Y-0.901
Y-0.889 X-0.858 X-0.822 X-0.796 Y-0.808 Y-0.834 Y-0.862 Y-0.884
Y-0.841 Y-0.844 X-0.822 Y-0.799 Y-0.8
G01
G01
X-0.72 X-0.801 X-0.844
G01
X-0.75 X-0.81 Y-0.859 Y-0.893 Y-0.899 Y-0.879 Y-0.86 X-0.9 X-0.944 X-0.966 Y-0.957 Y-0.915
Y-0.86 Y-0.811 Y-0.787 Y-0.782 Y-0.786 Y-0.795 Y-0.802 Y-0.818 Y-0.844 Y-0.871 Y-0.889 Y-0.885
Y-0.861 Y-0.828 X-0.802 X-0.805 X-0.833 Y-0.904 Y-0.914 Y-0.897 Y-0.729 Y-0.768 Y-0.784 Y-0.77
Y-0.709 Y-0.727 Y-0.744 Y-0.754 Y-0.762 Y-0.771 Y-0.776 Y-0.776
Y-0.745 Y-0.78 Y-0.818 Y-0.849 Y-0.86 Y-0.843 Y-0.803 Y-0.719
G01
G01 Y-0.843 Y-0.801
Y-0.757 Y-0.777 Y-0.781 Y-0.763 Y-0.769
G01 Y-0.74 Y-0.769 Y-0.79 Y-0.787 Y-0.762 Y-0.718 Y-0.713 Y-0.713 Y-0.72 Y-0.737 Y-0.759 Y-0.776
Y-0.786 Y-0.784 Y-0.762 Y-0.723
G01
G01
G01 Y-0.558 Y-0.532 Y-0.538 Y-0.546
```

Figura 15 Archivo .txt descargado

Para la creación del modelo de IA se tomaron en cuenta varias perspectivas, las cuales se listarán a continuación.

1 Aprendizaje por refuerzo

Debido a la complejidad que tomaba la programación del entorno se tomó la decisión de buscar una nueva forma para la generación del código G.

2 Redes Neuronales

Esta fue a una de las que más tiempo y recursos se le invirtió debido a que arrojaba datos muy positivos en la generación de código G, esto se muestra en Figura 16, Figura 17, Figura 18 y Figura 19, se dice que son muy positivos debido a que existen un momento en el que el accuracy llega a tener puntos más altos que el loss, por lo que se intuía que esto estaba funcionando de forma correcta, sin embargo esto cambio al probar las redes neuronales con datos que nunca había visto, después del análisis que se hizo a los resultados se concluyó que esta red neuronal estaba clasificando, lo cual fue el principal motivo por el que la red neuronal tuvo no podía generar código G.

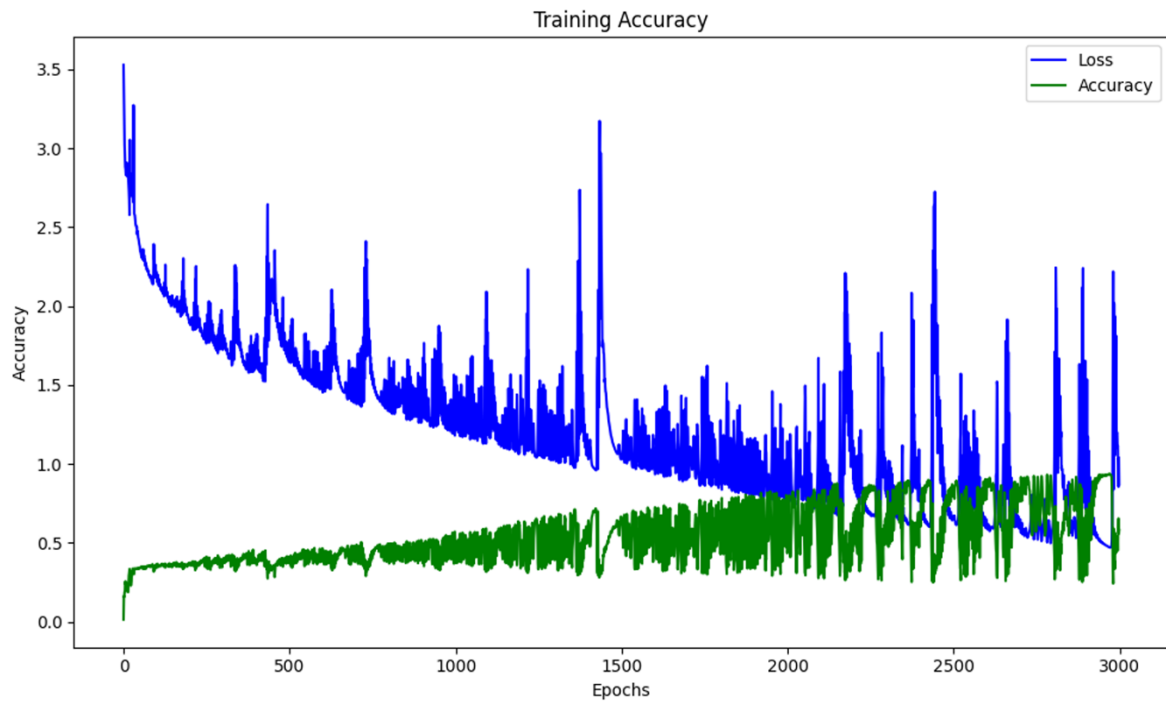


Figura 16 Grafica RN 1

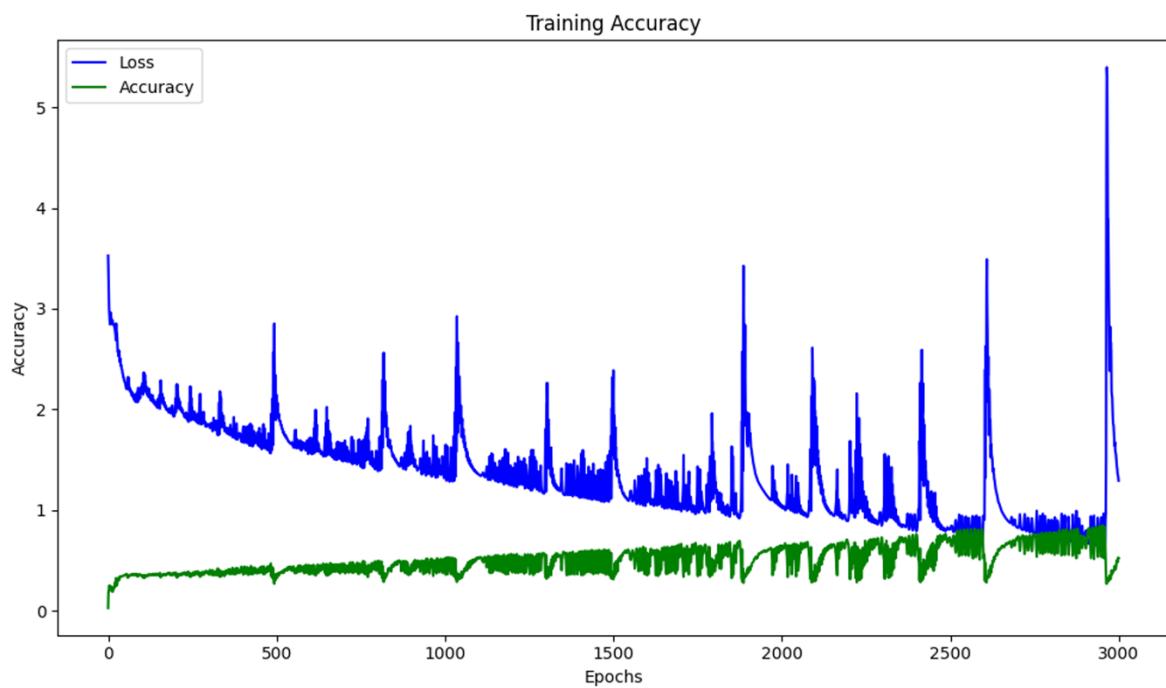


Figura 17 Grafica RN 2

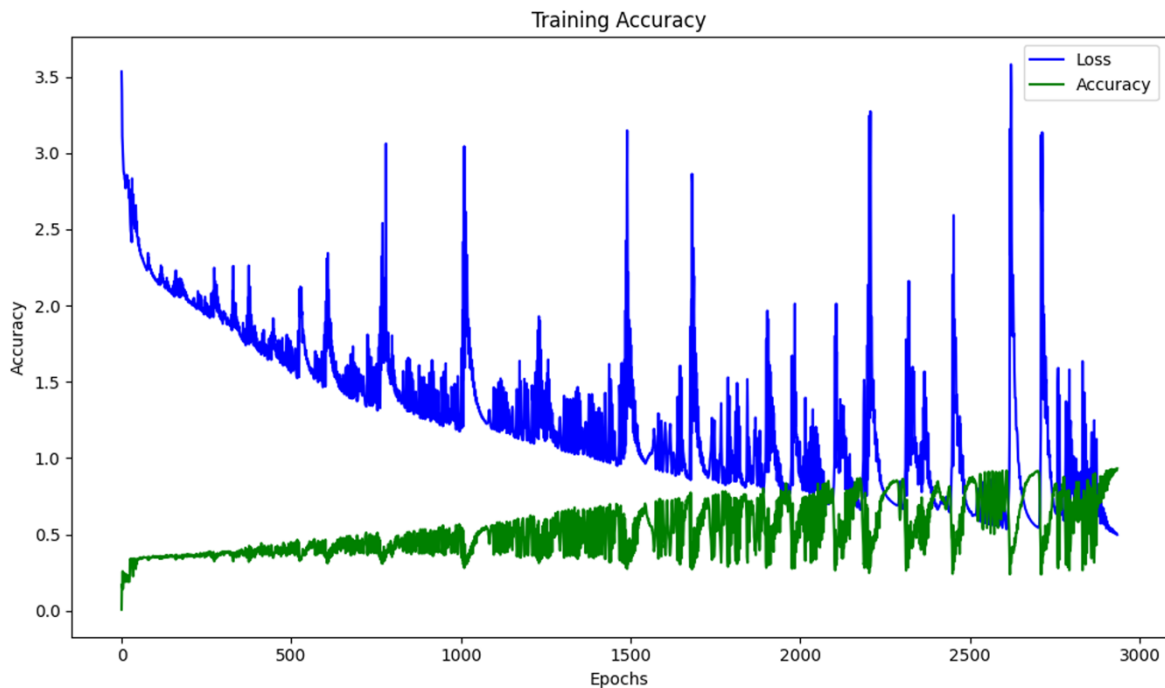


Figura 18 Grafica RN 3

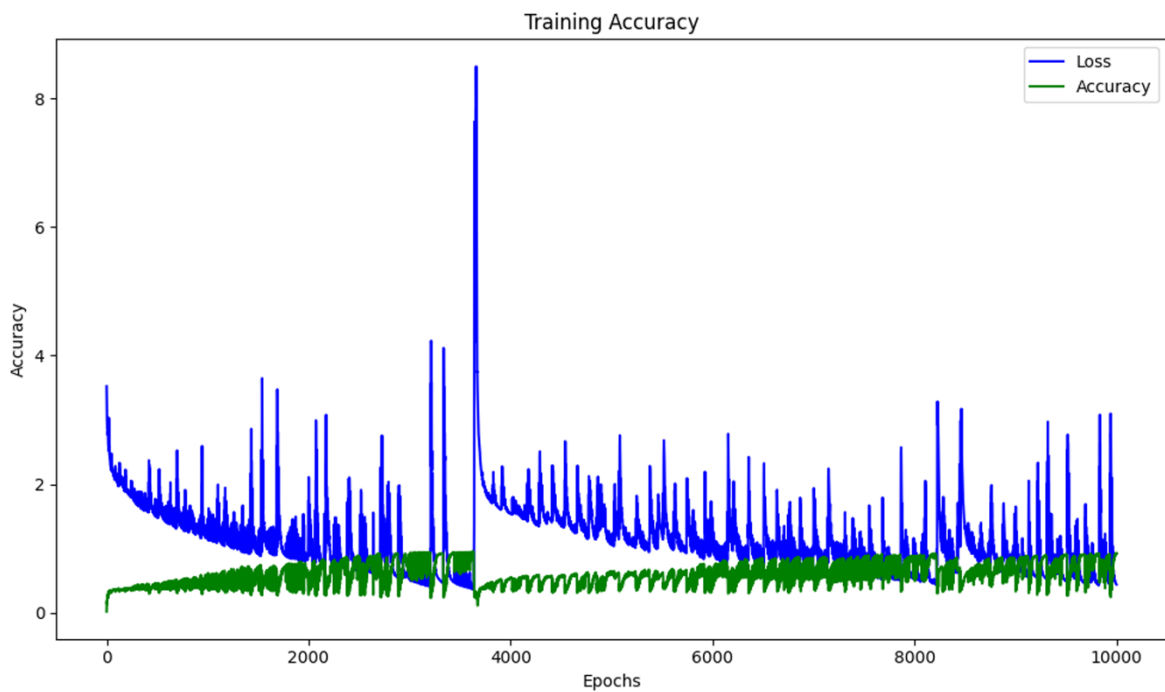


Figura 19 Grafica RN 4

3 Redes Generativas Adversas (GANs)

Se busco la implementación de estas redes debido a la capacidad que tienen para poder retroalimentarse, esto debido a la funcionabilidad del discriminador, pero debido a que este tipo de redes son muy complejas debido a los diversos errores de aprendizaje que existe, y por su compleja comprensión se decidió dejar de lado esta idea.

4 Transformers

Finalmente se llegó a la conclusión de que el uso de Transformers sería la mejor opción para el proyecto, esto debido a la popularidad que estas redes neuronales tienen para la traducción y la generación de textos, siendo uno de los más utilizados ChatGTP, estos Transformers están basados en técnicas de aprendizaje profundo y autoatención.

La principal desventaja en nuestro proyecto es la falta de un set de datos más amplio, esto es por la gran necesidad de los Transformers de alimentarse de datos, debido a este problema el entrenamiento del transformer es ineficiente, sin embargo, a pesar de las grandes cantidades de los que tiene, logra generar texto, el cual significa las bases de la IA.

La pérdida de las ultimas 10 épocas de entrenamiento se muestran en la [].

e. Seguimiento al plan de pruebas.

ID PRUEBA	ESTADO
EP 001	Aceptado
EP 002	Aceptado
EP 003	Se elimino la función
EP 004	Aceptado
EP 005	Se elimino la función
EP 006	Aceptado
EP 007	Aceptado
EP 008	Aceptado

La mayoría de las pruebas realizadas fueron aceptadas y cumplen con los criterios establecidos. Sin embargo, el EP 003 fue eliminado debido a la presencia de múltiples bugs durante la simulación. Como solución, se propuso que el sistema redirija hacia un simulador online para garantizar estabilidad y funcionalidad.

Por otro lado, el EP 005 también fue eliminado, ya que la funcionalidad de previsualización de archivos no pudo implementarse debido a la incompatibilidad de ciertas librerías con Flet, lo que imposibilitó su integración.

En general, las pruebas restantes se realizaron con éxito, alcanzando los resultados esperados.

f. Entrega o liberación.

Describir las actividades realizadas para la entrega y/o liberación formal del proyecto de trabajo terminal, se deberá incluir la evidencia correspondiente. En caso de existir incluir evidencia de la implementación del mismo.

- Análisis de costo

Conclusiones y Recomendaciones

El proyecto ha demostrado dar un paso sustancial en la generación automatizada de código G a través de inteligencia artificial, más precisamente con modelos basados en Transformers. Estos han demostrado un gran potencial en la decodificación de secuencias complejas con coherencia en los resultados, un proceso que se simplifica enormemente con respecto a lo que se solía obtener tradicionalmente.

Los principales problemas identificados durante el desarrollo fueron el principal desafío del conjunto de datos más grande y diverso para entrenar el modelo. Dado que los transformadores demandan abundantes datos para entrenar los modelos, su rendimiento se ve algo comprometido con conjuntos de datos de pequeño tamaño. Esto afectó en gran medida la eficiencia del entrenamiento y también la de las predicciones generadas por el modelo.

A pesar de esta limitación, los resultados obtenidos validan la viabilidad de esta metodología para la generación de código G. El modelo logró producir secuencias funcionales, estableciendo una base sólida para futuras mejoras.

Propuestas de mejora:

1. **Ampliación del conjunto de datos:** Se recomienda recopilar y procesar más datos relevantes para mejorar la capacidad de generalización del Transformer. Esto podría incluir una variedad de figuras geométricas y trayectorias más complejas.
2. **Optimización del modelo:** Ajustar hiperparámetros del Transformer y explorar variantes, como modelos preentrenados, para mejorar la eficiencia en entornos con datos limitados.
3. **Integración de simuladores avanzados:** Considerar el uso de simuladores más robustos para validar las predicciones generadas, reduciendo la dependencia de soluciones externas.
4. **Exploración de técnicas complementarias:** Investigar métodos de aprendizaje por refuerzo o redes generativas adversarias (GANs) para abordar problemas específicos que los Transformers puedan no resolver completamente.

Lecciones aprendidas:

- La recopilación de datos de alta calidad es esencial para el éxito de cualquier proyecto de inteligencia artificial.
- La flexibilidad en el diseño del sistema permitió realizar cambios efectivos, como la integración de un simulador online, para mitigar problemas técnicos.
- Los Transformers son una herramienta poderosa para tareas de generación, pero requieren una infraestructura adecuada y datos abundantes para maximizar su potencial.
- La asistencia constante a las clases y reuniones con el Scrum Master fue clave para la coordinación del equipo y la resolución de problemas, especialmente en etapas críticas del desarrollo. Mantener un flujo de comunicación activo con los responsables del proyecto permitió implementar soluciones rápidas y eficientes ante los desafíos que surgieron.

En conclusión, este proyecto sienta un precedente en la automatización de la generación de código G. Demuestra la capacidad de la inteligencia artificial para resolver problemas industriales. Si se implementan las mejoras propuestas, prevemos que este sistema se transformará en una solución más eficiente y flexible.

Fuentes de consulta.

- [1] A. Campos, V. Borja, y Á. Ayala, «Comparación de software shareware para generación de programas en código G», presentado en Memorias de Congreso: XXIII Congreso Internacional Anual de la SOMIM, Cuernavaca, Morelos, México, Septiembre. pp. MM142-149. ISSN, 2017, pp. 2448-5551.
- [2] O. J. Cifuentes Enciso, «Programación de Mecanizado en Código G con Macros».
- [3] G. R. V. Iscano, «Generador de código G con Python», 2022.
- [4] G. Kumar y P. K. Bhatia, «Impact of agile methodology on software development process», *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, vol. 2, n.º 4, pp. 46-50, 2012.
- [5] P. E. Montero, «Aprendizaje por refuerzo en espacios continuos», 2014.
- [6] E. Morales y J. González, «Aprendizaje por refuerzo», *Presentacion En Linea en: <https://ccc.inaoep.mx/~emorales/Cursos/Aprendizaje2/Acetatos/refuerzo.pdf>*, 2012.
- [7] Mordor Intelligence, «Mercado CNC - Tamaño, participación y crecimiento». [En línea]. Disponible en: <https://www.mordorintelligence.com/es/industry-reports/computer-numerical-controls-market>
- [8] P. P. Pande y Y. A. K. Kharche, «Development of Artificial Intelligence Algorithm for Automated CNC Machining Process for Unmanned Production», *Journal of Pharmaceutical Negative Results*, pp. 1392-1395, 2023.
- [9] S. Sachdeva, «Scrum methodology», *Int. J. Eng. Comput. Sci.*, vol. 5, n.º 16792, pp. 16792-16800, 2016.
- [10] E. Y. Zhang, A. D. Cheok, Z. Pan, J. Cai, y Y. Yan, «From turing to transformers: A comprehensive review and tutorial on the evolution and applications of generative transformer models», *Sci*, vol. 5, n.º 4, p. 46, 2023.

Apéndices

En caso de ser necesarios.