
PROYECTO 2 -IPC2

202307705 – Josué David Velásquez Ixchop

Resumen

El proyecto presentado se centra en el desarrollo de un software para simular el funcionamiento de una máquina ensambladora automática creada por la empresa Digital Intelligence, S.A. Dicha máquina utiliza líneas de ensamblaje y brazos robóticos para construir productos mediante la selección y ensamblaje de componentes. La novedad radica en su enfoque en optimizar el tiempo de ensamblaje, simulando cada paso del proceso. En el contexto nacional e internacional, este tipo de proyectos es relevante debido a la creciente automatización industrial.

Las posturas técnicas se enfocan en la implementación en Python, utilizando programación orientada a objetos y estructuras de datos abstractos (TDAs), evitando el uso de estructuras nativas como listas o diccionarios. A nivel económico y social, este tipo de tecnología puede incrementar la eficiencia productiva, mientras que su impacto ambiental se relaciona con la optimización de recursos.

Como conclusión, el ensayo destaca la importancia de aplicar principios de programación para resolver problemas de ensamblaje en tiempos óptimos, proporcionando una herramienta escalable y flexible para la industria.

Palabras clave

1. Automatización
2. Simulación
3. Ensamblaje
4. Tipos de Datos Abstractos
5. Líneas de producción

Abstract

The project focuses on the development of software to simulate the operation of an automatic assembly machine created by Digital Intelligence, S.A. This machine uses assembly lines and robotic arms to build products by selecting and assembling components. The innovation lies in its emphasis on optimizing assembly time by simulating each step of the process. Both nationally and internationally, this type of project is relevant due to the growing trend of industrial automation.

The technical approach centers on implementing Python, using object-oriented programming and abstract data types (ADTs), while avoiding the use of native structures like lists or dictionaries. Economically and socially, this technology can boost production efficiency, while its environmental impact is tied to resource optimization.

In conclusion, the essay highlights the importance of applying programming principles to solve assembly problems in optimal time, providing an industry-scalable and flexible tool.

Keywords

1. *Automation*
2. *Simulation*
3. *Assembly*
4. *Abstract Data Types*
5. *Production lines*

Introducción

El proyecto aborda el desarrollo de un software que simula el funcionamiento de una máquina ensambladora automática, diseñada por la empresa Digital Intelligence, S.A. En un contexto de creciente automatización industrial, este tema es fundamental, ya que la optimización de procesos productivos es clave para mejorar la eficiencia. La máquina emplea líneas de ensamblaje y brazos robóticos para seleccionar y ensamblar componentes de manera eficiente, lo que resalta su importancia en la producción moderna.

El ensayo explora la implementación de algoritmos en Python, con un enfoque en la programación orientada a objetos y el uso de Tipos de Datos Abstractos (TDAs). Se busca responder interrogantes clave: ¿Cómo optimizar el tiempo de ensamblaje de productos? ¿Qué estructuras de datos permiten una mayor eficiencia en este contexto?

El objetivo principal es demostrar cómo la tecnología puede mejorar la producción industrial, proporcionando una herramienta flexible y escalable para distintos sectores.

Desarrollo del tema

Este proyecto se orienta hacia la simulación de una máquina ensambladora automática, un desafío técnico significativo que busca optimizar los procesos de producción en entornos industriales contemporáneos. En la actualidad, la automatización se ha vuelto esencial para mejorar la eficiencia y reducir los costos en la manufactura, y este proyecto

se propone contribuir a esa tendencia mediante el desarrollo de una solución integral en Python. La creciente competencia en el sector industrial, impulsada por avances tecnológicos y la demanda de productos personalizados, exige que las empresas implementen sistemas más eficientes para optimizar su producción. Este proyecto no solo busca simular un proceso de ensamblaje, sino que también pretende servir como un modelo educativo y práctico que pueda ser adaptado y expandido en futuras aplicaciones industriales.

Para abordar este reto, se implementó una solución que hace uso de los principios de programación orientada a objetos (POO) y tipos de datos abstractos (TDA), además de herramientas de visualización como Graphviz, lo que permite no solo simular el ensamblaje, sino también visualizar de manera clara y efectiva el flujo del proceso. La POO se eligió como base del diseño debido a su capacidad para encapsular datos y comportamientos en clases, lo que facilita la creación de un sistema modular y escalable. Esto permite que cada componente del sistema sea desarrollado y mantenido de manera independiente, promoviendo la reutilización de código y simplificando futuras actualizaciones.

Clases Fundamentales del Proyecto

El diseño del sistema se basa en varias clases fundamentales que gestionan las estructuras de datos y las operaciones necesarias para simular el ensamblaje de productos de manera eficiente.

Clase CustomList

La clase **CustomList** se desarrolló como una estructura de datos personalizada para gestionar de manera eficiente las máquinas y productos en el sistema. Esta clase permite realizar operaciones de inserción, eliminación y acceso a elementos, lo que

resulta fundamental para mantener un seguimiento preciso del estado de cada máquina y sus productos asociados. A diferencia de las listas nativas de Python, que pueden presentar limitaciones en cuanto a la flexibilidad y la personalización, **CustomList** está diseñada específicamente para el contexto del proyecto, lo que asegura un rendimiento óptimo. Los métodos implementados, como `add(element)`, que añade un nuevo elemento a la lista, y `get(index)`, que retorna el elemento en la posición especificada, facilitan la manipulación de los datos a lo largo del proceso de ensamblaje. Además, se incluye un método `size()` que devuelve el número total de elementos en la lista, lo cual es crucial para la gestión de iteraciones y condiciones en el flujo de trabajo.

Clase ComponenteContador

La clase **ComponenteContador** se diseñó específicamente para encapsular la información relativa a cada componente del producto, incluyendo su nombre y la cantidad necesaria para completar el ensamblaje. Esta clase es crucial para el seguimiento de los recursos en el proceso de producción, ya que permite incrementar la cantidad de un componente cada vez que se completa un ensamblaje. Esto no solo facilita una gestión precisa de los insumos, garantizando que se disponga de la cantidad necesaria para cumplir con la demanda de producción, sino que también ayuda a optimizar el uso de recursos, evitando desperdicios y mejorando la sostenibilidad del proceso de manufactura. Esta clase incluye métodos para incrementar la cantidad, así como para obtener una representación legible del componente, lo que resulta útil para la generación de reportes.

Clase Resultado

Además, la clase **Resultado** fue implementada para representar el estado de la simulación en cada instante

de tiempo. Esta clase almacena las acciones realizadas por los brazos robóticos en cada línea de ensamblaje, permitiendo un seguimiento detallado del progreso del ensamblaje en tiempo real. Almacenar el resultado de cada acción en instancias de esta clase proporciona una base sólida para la generación de reportes y la visualización del proceso, ofreciendo una perspectiva clara de cómo se desarrolló la producción a lo largo del tiempo. La inclusión de métodos que permiten agregar las acciones realizadas y obtener un resumen del estado en cada momento es fundamental para la comprensión del flujo de trabajo y para identificar posibles cuellos de botella o ineficiencias en el proceso de ensamblaje.

Funciones Clave del Proyecto

En cuanto a las funcionalidades del programa, la función principal, **construir**, se encarga de simular el ensamblaje de un producto específico. Esta función es esencial, ya que procesa la configuración de la máquina y los productos seleccionados por el usuario, calculando el tiempo total necesario para completar el ensamblaje. Durante esta simulación, se registran las acciones de los brazos robóticos y el estado de cada componente, lo que permite evaluar la eficiencia del proceso de ensamblaje y detectar posibles áreas de mejora. A medida que se avanza en la simulación, se actualizan las instancias de la clase **Resultado**, proporcionando un registro detallado de las acciones y el tiempo utilizado, lo que es vital para el análisis posterior.

Por otra parte, la función **generate_assembly_graph** se implementó para crear gráficos visuales que representan los pasos pendientes del ensamblaje. Utilizando Graphviz, esta función permite al usuario visualizar el flujo de trabajo, identificar cuellos de

botella y entender mejor la secuencia de ensamblaje, lo que puede ser invaluable para optimizar el proceso en el futuro. La capacidad de visualizar las interacciones y los pasos pendientes en un formato gráfico no solo mejora la comprensión del sistema, sino que también permite a los usuarios identificar áreas de mejora y optimización en sus procesos de producción.

Generación de Reportes

Al final de la simulación, se genera un archivo XML que contiene todos los resultados del proceso de ensamblaje, incluyendo el tiempo total y los pasos realizados. Esta funcionalidad de generar un reporte estructurado no solo permite un análisis posterior, sino que también facilita la toma de decisiones en la gestión de la producción. Utilizando la biblioteca `xml.etree.ElementTree`, se construye una estructura XML que incluye información sobre el tiempo de ensamblaje y los pasos realizados, lo que asegura que los resultados sean fácilmente accesibles y utilizables para diversas finalidades, como la revisión y la presentación a los stakeholders. Además, se proporciona la opción de descargar reportes en formatos HTML o XML, lo que asegura que la información generada sea accesible y útil para el análisis de los procesos de producción.

Interacción del Usuario y Experiencia

El sistema también fue diseñado teniendo en cuenta la interacción del usuario y la facilidad de uso. Se implementó una interfaz web utilizando Flask, lo que permite a los usuarios cargar archivos XML con la configuración de máquinas y productos, seleccionar la máquina y el producto que desean simular, y finalmente visualizar los resultados de la simulación. Esta interfaz intuitiva facilita la interacción con el sistema, permitiendo a los usuarios realizar

simulaciones sin necesidad de tener conocimientos técnicos avanzados. La disposición clara de las opciones y los resultados proporciona una experiencia de usuario fluida, lo que es fundamental para fomentar la adopción y el uso eficaz del sistema en entornos industriales.

Impacto y Futuras Direcciones

En resumen, este proyecto proporciona una solución integral para la simulación de una máquina ensambladora automática, abordando todos los aspectos críticos del proceso de ensamblaje mediante el uso de clases bien definidas, estructuras de datos eficientes y funcionalidades robustas. La implementación de este sistema no solo contribuye a la comprensión de los procesos de producción automatizados, sino que también establece una base sólida para futuras investigaciones y desarrollos en el ámbito de la automatización industrial. A medida que la tecnología avanza y las demandas del mercado cambian, este sistema puede expandirse y adaptarse para incorporar nuevas funcionalidades, como la integración con sistemas de gestión de producción, el análisis de datos en tiempo real y la optimización basada en inteligencia artificial.

La capacidad del proyecto para adaptarse a diferentes configuraciones de máquinas y productos, así como su enfoque modular, garantiza que puede ser utilizado como un modelo práctico para empresas que buscan mejorar su eficiencia operativa y optimizar sus procesos de producción. Con el tiempo, este sistema podría convertirse en una herramienta valiosa para la formación y la capacitación en automatización industrial, así como un recurso clave para ingenieros y técnicos que trabajan en el diseño y la implementación de soluciones de ensamblaje automatizadas en la industria.

Conclusiones

El presente proyecto ha demostrado la viabilidad de simular el funcionamiento de una máquina ensambladora automática, utilizando herramientas de programación moderna y enfoques de desarrollo que favorecen la eficiencia y la escalabilidad. A través de la implementación de clases y estructuras de datos personalizadas, se ha conseguido crear un sistema que no solo simula el proceso de ensamblaje, sino que también proporciona información valiosa sobre el rendimiento y la optimización de recursos. Este enfoque permite a las empresas industriales evaluar su producción y detectar áreas de mejora, promoviendo una mayor competitividad en un mercado en constante evolución.

Uno de los aportes más destacados de este trabajo es la integración de visualización gráfica mediante Graphviz, lo cual facilita la comprensión del flujo de trabajo y permite a los usuarios identificar rápidamente cuellos de botella en el proceso de producción. Esta visualización es fundamental para la toma de decisiones informadas y la implementación de mejoras en tiempo real. Además, la capacidad de generar reportes estructurados en formatos accesibles, como XML y HTML, asegura que la información generada sea útil para diversas audiencias, desde operadores de planta hasta gerentes de producción.

Sin embargo, quedan abiertas varias preguntas para la reflexión y el debate en torno a la automatización industrial. ¿Cómo pueden las empresas equilibrar la necesidad de automatización con el impacto social de la reducción de empleos en ciertos sectores? ¿Qué papel jugarán las nuevas tecnologías, como la

inteligencia artificial y el aprendizaje automático, en la optimización de procesos de ensamblaje en el futuro? Además, es crucial considerar la sostenibilidad en la automatización; ¿cómo pueden las empresas implementar tecnologías que no solo sean eficientes, sino también respetuosas con el medio ambiente?

Para profundizar en esta temática, se recomienda la exploración de áreas como el diseño de algoritmos avanzados que incorporen inteligencia artificial para mejorar la predicción de tiempos de ensamblaje y la gestión de recursos. Asimismo, investigar la integración de sistemas de automatización con Internet de las Cosas (IoT) podría ofrecer nuevas oportunidades para el monitoreo en tiempo real de las líneas de producción, brindando datos valiosos que permitan la optimización continua del proceso.

En conclusión, este proyecto no solo ofrece una herramienta práctica para simular el ensamblaje automatizado, sino que también abre la puerta a un amplio campo de investigación y desarrollo en la automatización industrial. La flexibilidad del sistema y su capacidad para adaptarse a diversas configuraciones representan un avance significativo en la comprensión y mejora de los procesos productivos.

Referencias bibliográficas

1. GAC Formación. (n.d.). *S15: Conceptualización y evaluación de TDAs*. GAC Formación. <https://gacformacion.com/producto/s15-conceptualizacion-y-evaluacion-de-tdas/>
2. J2Logo. (n.d.). *Tutorial Flask en español*. J2Logo. <https://j2logo.com/tutorial-flask-espanol/>

Apéndices

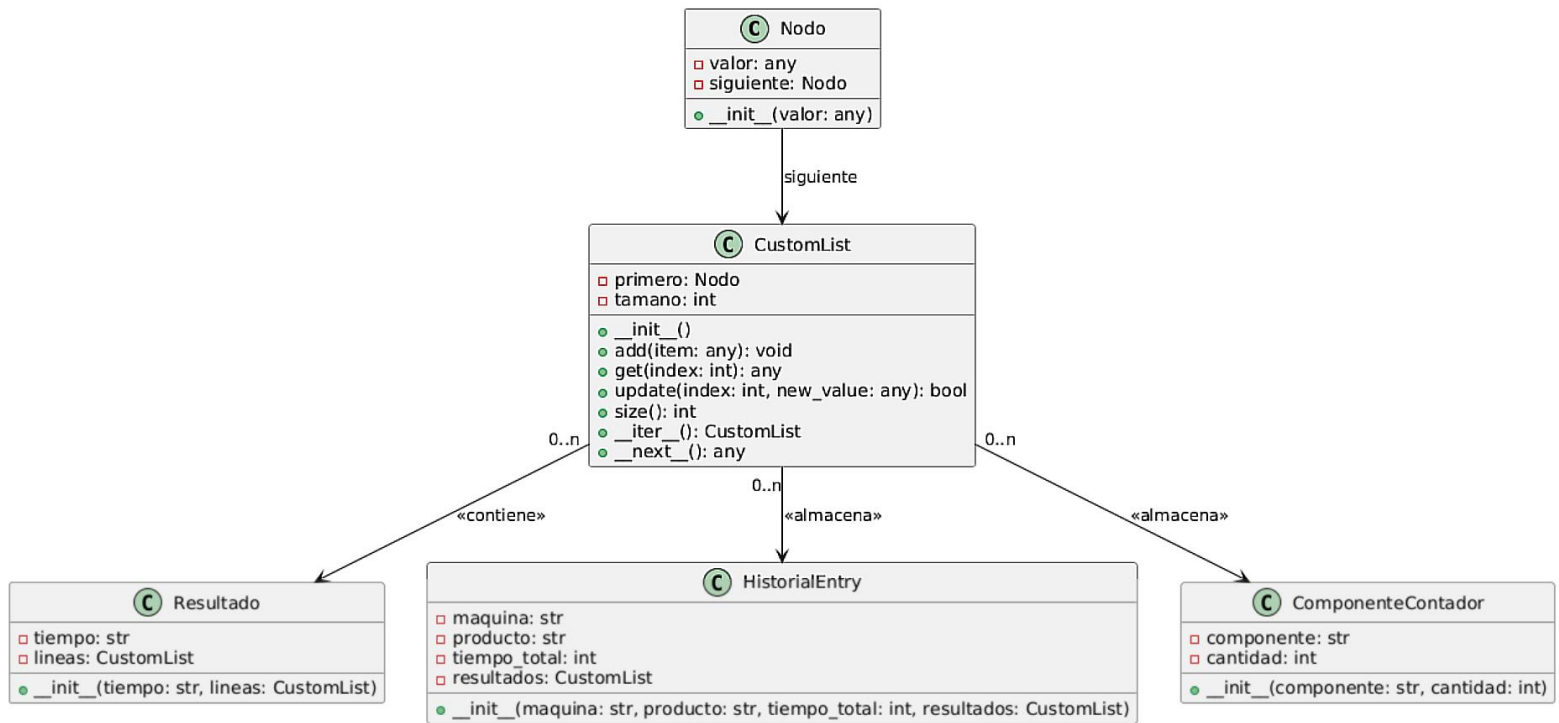


Figura 1. Diagrama de Clase
Fuente: elaboración propia

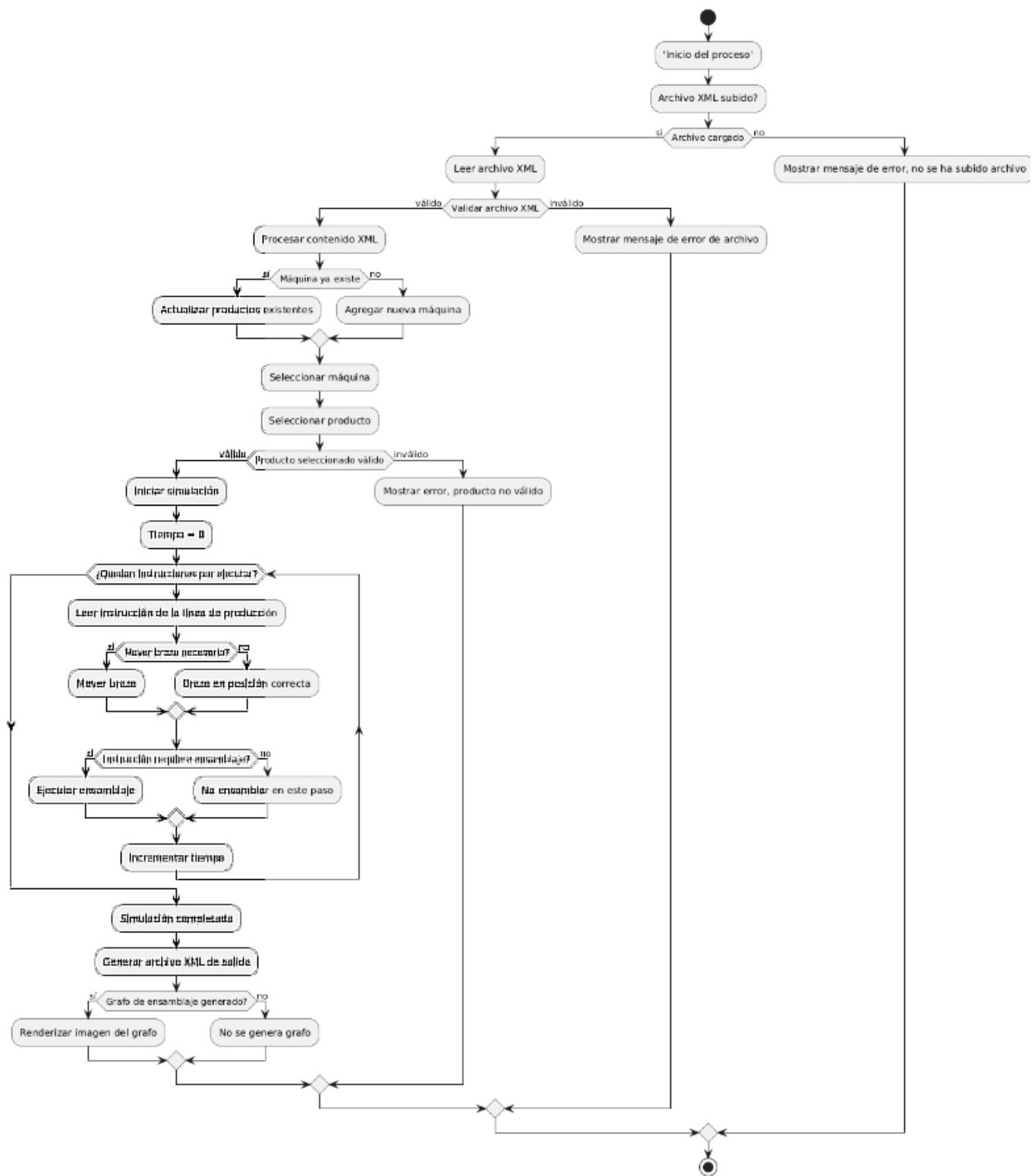


Figura 2. Diagrama de Actividades
Fuente: elaboración propia

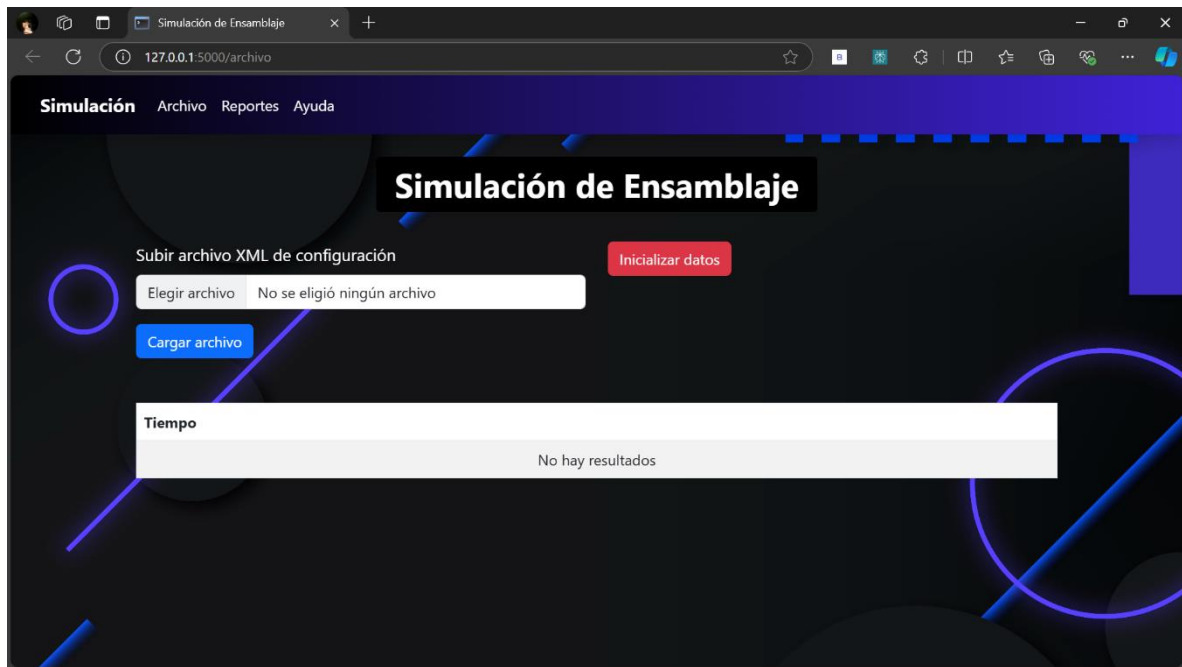


Figura 3. Página Principal
Fuente: elaboración propia

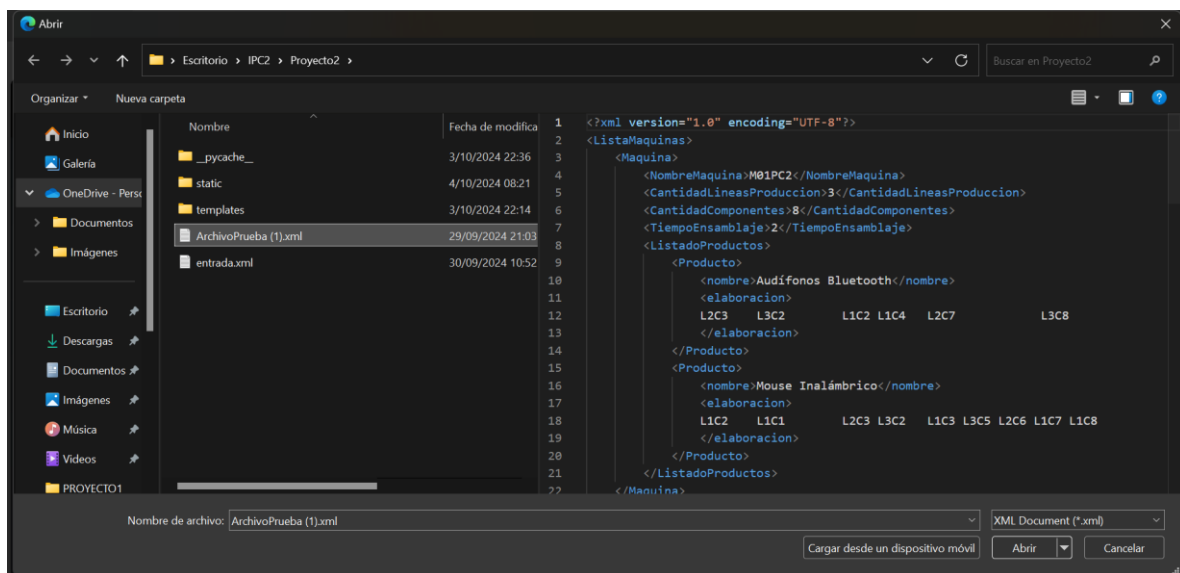


Figura 4. Carga XML
Fuente: elaboración propia

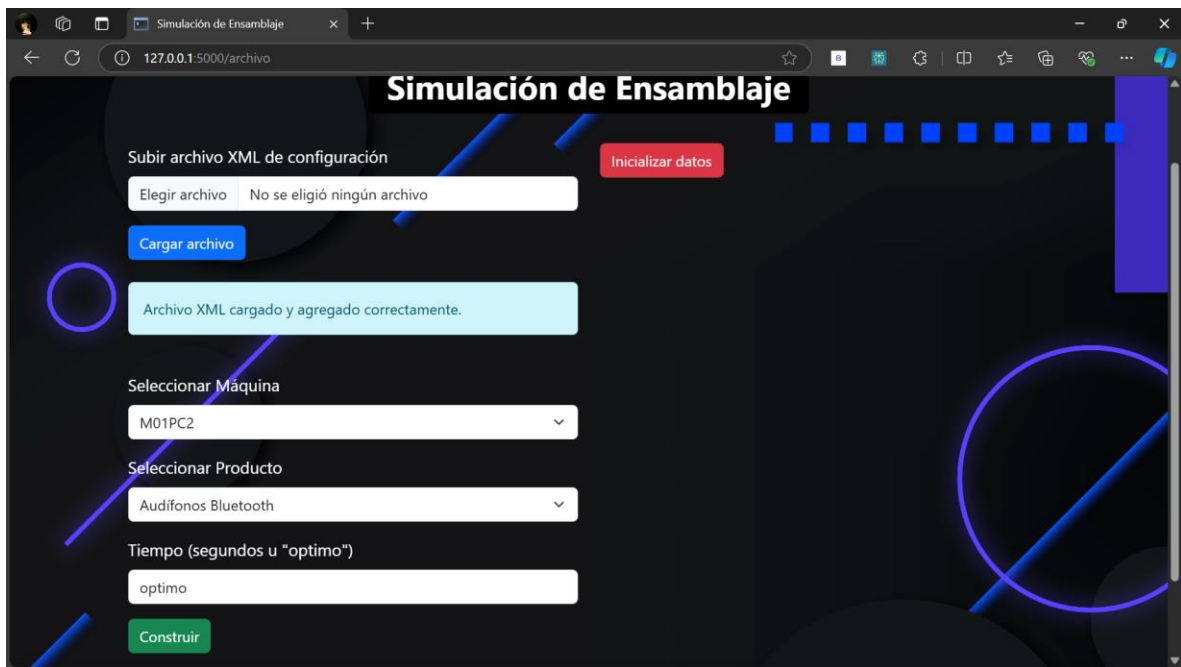


Figura 5. Selección de maquinas y productos XML
Fuente: elaboración propia

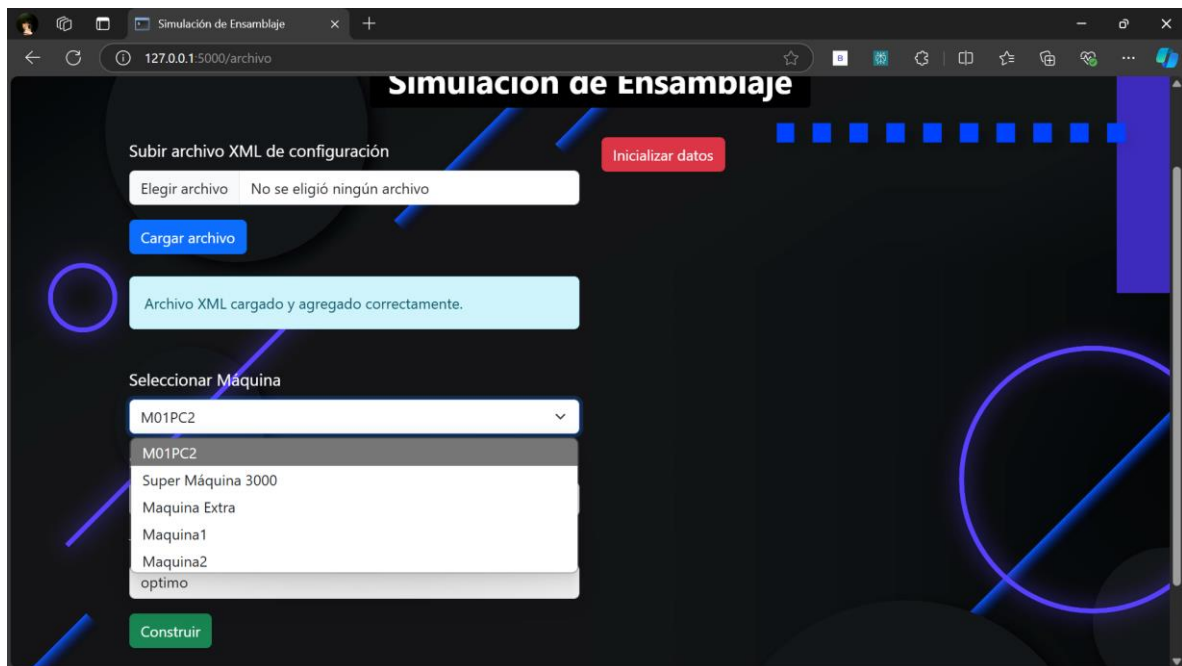


Figura 6. Sobrecarga de XML de entrada
Fuente: elaboración propia

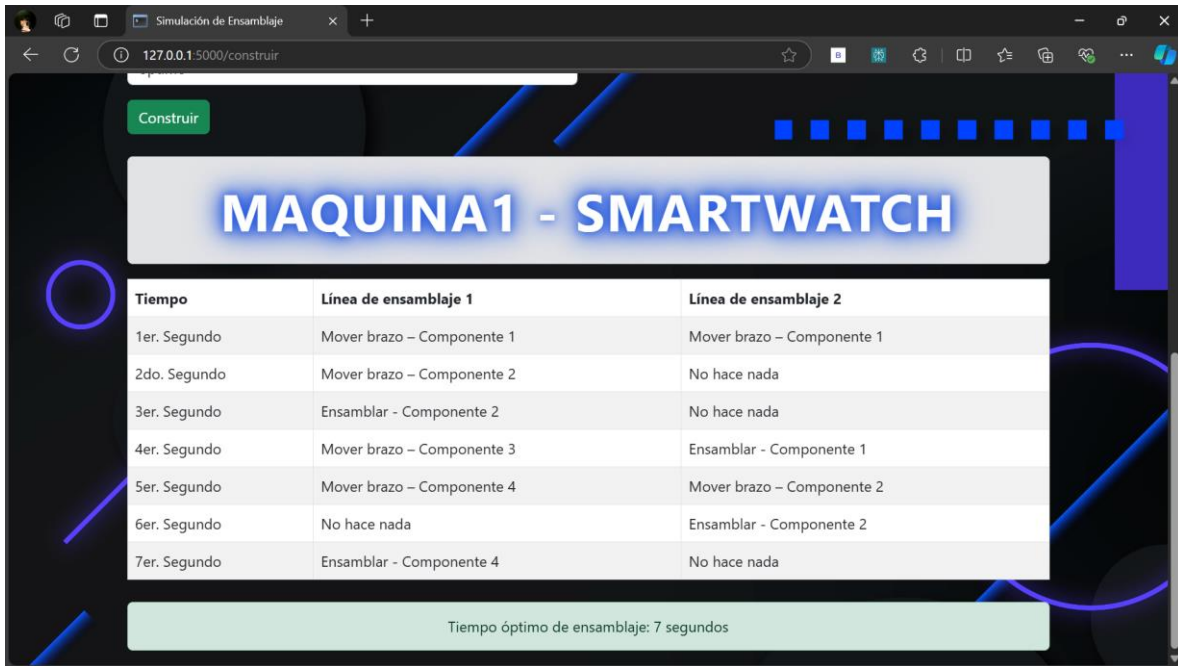


Figura 7. Tablas de pasos optima
Fuente: elaboración propia

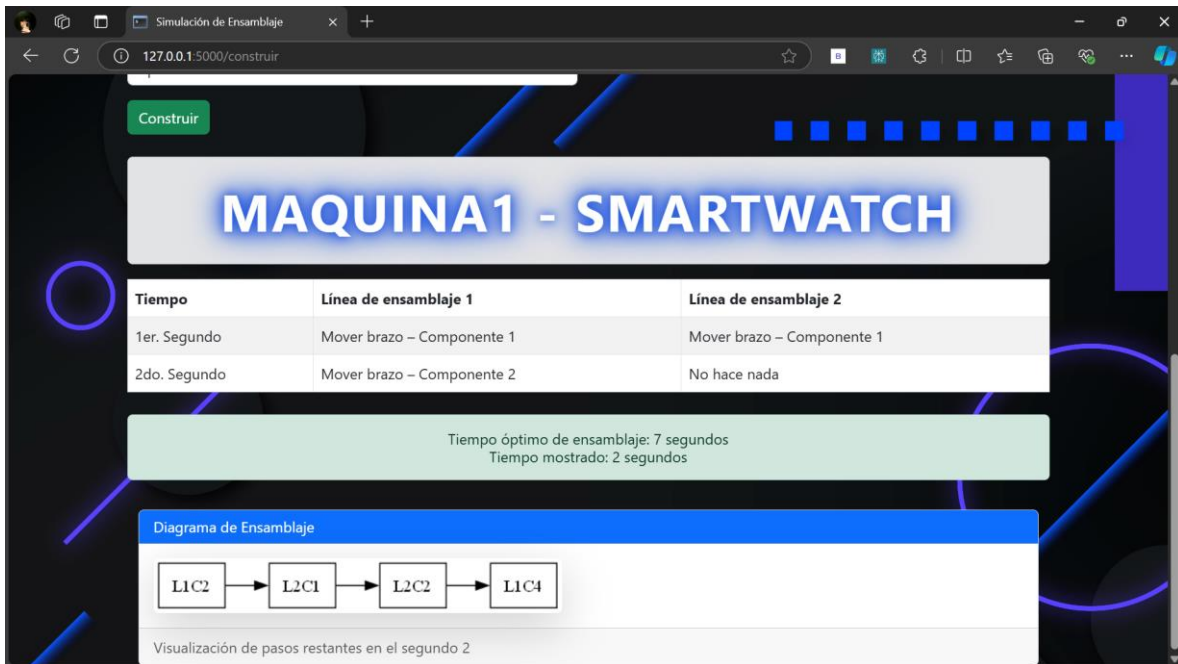


Figura 8. Tablas de pasos en un tiempo “t”
Fuente: elaboración propia

Reportes de Ensamblaje

Reporte #1

Máquina: M01PC2
Producto: Audífonos Bluetooth

Tiempo Total: 17 segundos

Tiempo	Línea 1	Línea 2	Li
1er. Segundo	Mover brazo	Mover brazo	M
	Componente 1	Componente 1	C
2do. Segundo	Mover brazo	Mover brazo	M

Descargar Reporte HTML

Reporte #2

Máquina: M01PC2
Producto: Audífonos Bluetooth

Tiempo Total: 17 segundos

Tiempo	Línea 1	Línea 2	Li
1er. Segundo	Mover brazo	Mover brazo	M
	Componente 1	Componente 1	C
2do. Segundo	Mover brazo	Mover brazo	M

Descargar XML de Salida

Figura 9. Reportes html de pasos
Fuente: elaboración propia

Centro de Ayuda

Encuentra aquí la información sobre la aplicación y el desarrollador.

Información del Estudiante

Nombre: Josué David Velásquez Ixchop
Carnet: 202307705
Curso: IPC2
Sección: A

Acerca de la Aplicación

Esta aplicación permite simular el ensamblaje de productos en diversas máquinas configuradas mediante archivos XML.
Versión: 1.0.0

Documentación

Consulta la documentación oficial para obtener más detalles sobre cómo utilizar la aplicación.

[Ver Documentación](#)

Figura 10. Sección de ayuda al usuario
Fuente: elaboración propia

```
static > simulacion_salida.xml
1 <?xml version="1.0"?>
2 <SalidaSimulacion>
3   <Maquina>
4     <Nombre>M81PC2</Nombre>
5     <ListadoProductos>
6       <Producto>
7         <Nombre>Audifonos Bluetooth</Nombre>
8         <TiempoTotal>17</TiempoTotal>
9       <ElaboracionOptima>
10        <Tiempo NoSegundo="1">
11          <LineaEnsamblaje NoLinea="1">Mover brazo Componente 1</LineaEnsamblaje>
12          <LineaEnsamblaje NoLinea="2">Mover brazo Componente 1</LineaEnsamblaje>
13          <LineaEnsamblaje NoLinea="3">Mover brazo Componente 1</LineaEnsamblaje>
14        </Tiempo>
15        <Tiempo NoSegundo="2">
16          <LineaEnsamblaje NoLinea="1">Mover brazo Componente 2</LineaEnsamblaje>
17          <LineaEnsamblaje NoLinea="2">Mover brazo Componente 2</LineaEnsamblaje>
18          <LineaEnsamblaje NoLinea="3">Mover brazo Componente 2</LineaEnsamblaje>
19        </Tiempo>
20        <Tiempo NoSegundo="3">
21          <LineaEnsamblaje NoLinea="1">No hace nada</LineaEnsamblaje>
22          <LineaEnsamblaje NoLinea="2">Mover brazo Componente 3</LineaEnsamblaje>
23          <LineaEnsamblaje NoLinea="3">No hace nada</LineaEnsamblaje>
24        </Tiempo>
25        <Tiempo NoSegundo="4">
26          <LineaEnsamblaje NoLinea="1">No hace nada</LineaEnsamblaje>
27          <LineaEnsamblaje NoLinea="2">Ensamblar - Componente 3</LineaEnsamblaje>
28          <LineaEnsamblaje NoLinea="3">No hace nada</LineaEnsamblaje>
29        </Tiempo>
30        <Tiempo NoSegundo="5">
31          <LineaEnsamblaje NoLinea="1">No hace nada</LineaEnsamblaje>
32          <LineaEnsamblaje NoLinea="2">Ensamblar - Componente 3</LineaEnsamblaje>
```

Figura 11. Generación de archivo de salida
Fuente: elaboración propia