

AUTOMATIZACION DE ORDENES (Máximo 25 palabras)

(Eliminar todo el contenido entre paréntesis al momento de la entrega)

TÍTULO EN INGLÉS EN MAYÚSCULAS(ORDER AUTOMATION)

Ensayo

(Elegir cualquiera de las 3 tipologías)

Del Valle, Daniel

(Universidad del Valle de Puebla)

(Correo electrónico)

ORCID: (<https://orcid.org/>)

García, Ever

(Universidad del Valle de Puebla)

(Correo electrónico)

ORCID:

Guzmán, Austin

(Universidad del Valle de Puebla)

(Correo electrónico)

ORCID: (<https://orcid.org/>)

(Máximo 4 autores)

Reseña de Daniel del Valle 1

Como aportaciones importantes desarrollar los objetivos específicos y generales dentro de la evolución del proyecto.

Reseña de Ever García 2

Como aportaciones y complemento crear el cronograma de actividades así como la investigación de el estado del arte.

Reseña de Austin Bautista 2

Como aportaciones la creación del presupuesto total que se llevara con la creación del programa.

Resumen

(Este código en C# ayudaría a la fábrica de ónix a automatizar su proceso de generación de órdenes de producción, lo que podría resultar en una mayor eficiencia y productividad. Sin embargo, la implementación de tal sistema requeriría una comprensión sólida de la programación en C# y de los procesos de producción de la fábrica.)

Palabras clave: (5 palabras clave utilizando el Thesaurus de la UNESCO, separados por comas.)

(<http://vocabularies.unesco.org/browser/thesaurus/es/>)

Abstract

(250 palabras como máximo colocando los siguientes elementos: planteamiento del problema, metodología, principales resultados y conclusiones. Se recomienda la revisión del Abstract al utilizar traductores a fin de tener coherencia en la lectura).

Keywords: (5 palabras clave utilizando el Thesaurus de la UNESCO en inglés, separado por comas.)

Introducción

El objetivo principal de este código sería automatizar el proceso de generación de órdenes de producción en una fábrica de ónix. Esto implicaría la creación de un sistema que pueda interactuar con la base de datos de la fábrica, recoger los detalles necesarios y generar una orden de producción basada en ciertos criterios. El código en C# se elegiría por su robustez y su amplio uso en el desarrollo de aplicaciones empresariales. C# es un lenguaje de programación orientado a objetos que es altamente versátil y puede interactuar con una variedad de bases de datos, lo que lo hace ideal para este proyecto. El sistema tendría que ser capaz de realizar varias tareas, como recoger los detalles de los productos a fabricar, determinar la cantidad

necesaria, calcular el tiempo de producción y asignar los recursos necesarios. Todo esto se haría de acuerdo con las reglas y regulaciones establecidas por la fábrica.

Planteamiento del problema

La problemática a abordar en la fábrica de ónix podría ser la siguiente:

La fábrica de ónix actualmente genera sus órdenes de producción de manera manual, lo que puede llevar a errores humanos, retrasos y una eficiencia general reducida. Este proceso manual también puede limitar la capacidad de la fábrica para adaptarse rápidamente a los cambios en la demanda del mercado. Además, la falta de automatización puede resultar en un uso ineficiente de los recursos, ya que la fábrica puede no ser capaz de optimizar completamente su producción en función de los recursos disponibles.

Por lo tanto, el problema que se plantea es cómo mejorar la eficiencia y la productividad de la fábrica de ónix mediante la automatización del proceso de generación de órdenes de producción. La solución propuesta es desarrollar un código en C# que pueda interactuar con la base de datos de la fábrica y generar automáticamente órdenes de producción basadas en ciertos criterios. Esto permitiría a la fábrica optimizar su producción, reducir errores y adaptarse más rápidamente a los cambios en la demanda del mercado.

Estado del Arte.

Los programas cliente que automatizan Office pueden crear una nueva instancia de ese programa de Office u obtener una referencia a la instancia que ya se está ejecutando. Por lo general, Microsoft recomienda crear una nueva instancia en lugar de asociarla a una instancia en ejecución. Sin embargo, en algunos escenarios, el programa cliente debe automatizar una instancia ya en ejecución de un programa de Office. En este caso, el cliente de Automación obtiene la referencia al objeto Del modelo de objetos componentes (COM) del servidor de Automación de la tabla de objetos en ejecución (ROT). Si el servidor de Automación se registró en la tabla de objetos en ejecución, un cliente de .NET puede obtener una referencia a la instancia en ejecución. Los servidores COM se pueden clasificar como Uso múltiple (instancia única) o Uso único (varias instancias), en función del número de instancias de ese servidor que se pueden ejecutar simultáneamente en un único equipo.

Cuando una solicitud para un nuevo objeto COM llega a un servidor COM multiuso (instancia única), el servidor usa solo una instancia del archivo .exe para crear ese objeto. Independientemente del número de clientes que soliciten un nuevo objeto COM, solo habrá un servidor.exe proceso. En el servidor de uso único (varias instancias), cada solicitud de un nuevo objeto COM inicia una instancia independiente del servidor.exe archivo. Por lo tanto, más de una instancia del servidor se puede ejecutar en el mismo equipo.

Fuente: <https://learn.microsoft.com/es-es/previous-versions/office/troubleshoot/office-developer/use-visual-c-automate-run-program-instance> .)

Objetivo General

° Crear un código en c# que genere automáticamente ordenes de producción para una fábrica de ónix y acelerar sus procesos.

Objetivos Específicos

Definir los parámetros de las órdenes de producción: Esto puede incluir detalles como el tipo de producto, la cantidad, la fecha de entrega, etc.

Establecer las reglas para la generación automática de órdenes: Por ejemplo, se pueden establecer reglas para determinar cuándo se debe generar una nueva orden de producción (por ejemplo, cuando el inventario de un producto cae por debajo de un cierto nivel).

Integrar el sistema con la base de datos de la fábrica: El sistema debe ser capaz de acceder a la base de datos de la fábrica para obtener la información necesaria para generar las órdenes de producción.

Desarrollar una interfaz de usuario amigable: La interfaz de usuario debe ser fácil de usar para que los empleados de la fábrica puedan interactuar con el sistema de manera eficiente.

Implementar medidas de seguridad: Para proteger la información de la fábrica, el sistema debe tener medidas de seguridad adecuadas, como el cifrado de datos y la autenticación de usuarios.

Proporcionar funcionalidad para el seguimiento de órdenes: El sistema debe permitir a los usuarios rastrear el estado de las órdenes de producción.

Incluir funcionalidad para la generación de informes: El sistema debe ser capaz de generar informes que proporcionen información útil sobre el proceso de producción, como el número de órdenes generadas, el tiempo medio de producción, etc.

Estos objetivos específicos ayudarán a asegurar que el sistema cumpla con las necesidades de la fábrica y mejore su proceso de producción

Método y Metodología

(Análisis de requisitos: Comprender las necesidades y requisitos específicos fábrica. Esto puede incluir la comprensión de cómo se generan actualmente las órdenes de producción, qué información se requiere en estas órdenes, y cualquier regla o restricción específica que deba seguir el sistema.

Diseño del sistema: Basándose en los requisitos, diseñar un sistema que pueda automatizar el proceso de generación de órdenes de producción. Esto puede incluir la

definición de cómo se almacenará y gestionará la información, cómo se generarán las órdenes de producción, y cómo se manejarán las excepciones o errores.

Desarrollo del código: Utilizando C#, desarrollar el código que implementa el diseño del sistema. Esto puede incluir la escritura de funciones para generar órdenes de producción, manejar errores, y realizar cualquier otra tarea necesaria.

Pruebas: Una vez que el código esté desarrollado, realizar pruebas para asegurarse de que el sistema funciona como se esperaba. Esto puede incluir la realización de pruebas unitarias, pruebas de integración y pruebas de sistema.

Implementación: Después de que el sistema ha sido probado y se ha confirmado que funciona correctamente, puede ser implementado en la fábrica. Esto puede incluir la formación de los empleados sobre cómo utilizar el nuevo sistema, y la monitorización del sistema para asegurarse de que está funcionando correctamente.

Mantenimiento y actualizaciones: Una vez que el sistema esté en funcionamiento, puede ser necesario realizar mantenimiento y actualizaciones periódicas para asegurarse de que sigue satisfaciendo las necesidades de la fábrica

)

Conclusiones y discusión

(Crear un programa de automatización de órdenes en C# para una fábrica de ónix podría tener varias conclusiones positivas:

Eficiencia: La automatización puede mejorar la eficiencia al reducir el tiempo necesario para procesar las órdenes y minimizar los errores humanos.

Ahorro de costos: Aunque la implementación inicial puede requerir una inversión, a largo plazo, la automatización puede resultar en un ahorro de costos significativo.

Mejora en la calidad del servicio al cliente: Con la automatización, las órdenes pueden ser procesadas de manera más rápida y precisa, lo que puede llevar a una mejor experiencia para el cliente.

Análisis de datos: Un programa automatizado puede recopilar y analizar datos de manera más eficiente, lo que puede proporcionar información valiosa para la toma de decisiones empresariales.

Escalabilidad: Un sistema automatizado puede manejar un aumento en el volumen de órdenes sin la necesidad de contratar más personal.

.)

Referencias

(El modelo de citas y referencias utilizado para la presentación de proyectos para las revistas de la Universidad del Valle de Puebla es el APA 7ma. Edición. Recordando que todos las citas colocadas dentro del cuerpo del documento deben de estar correctamente referenciadas en este apartado.

Para conocer más profundamente sobre el modelo APA se puede visitar este sitio: <https://normas-apa.org/wp-content/uploads/Guia-Normas-APA-7ma-edicion.pdf>)

Anexos

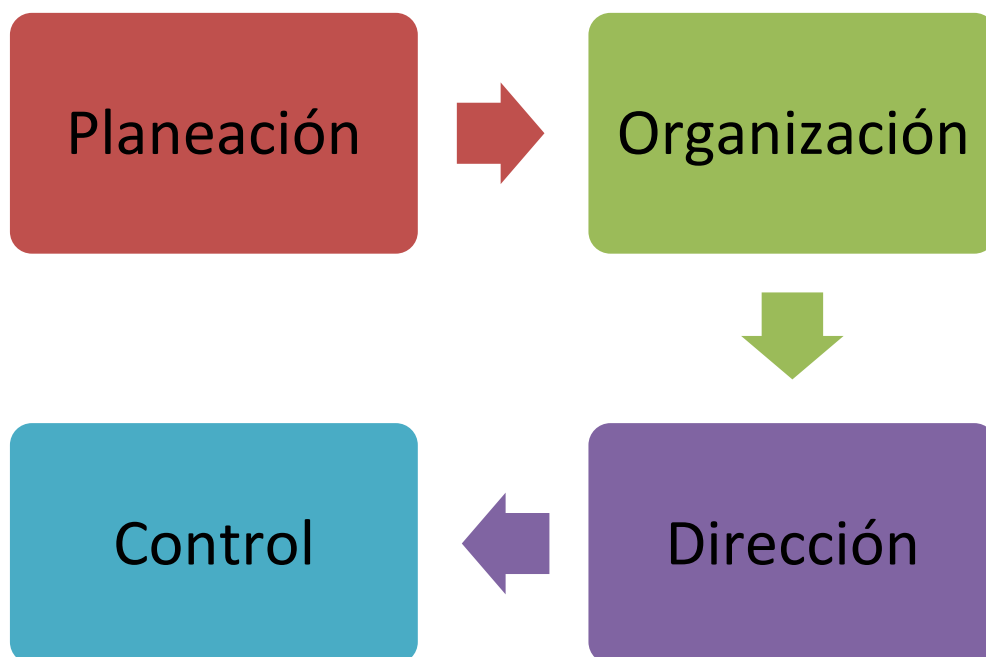
(Los anexos se colocan en esa sección de ser necesario y fungen como un recurso para no interrumpir la narración del texto. Aquí se pueden colocar instrumentos completos, resultados de encuestas entre otros. Los anexos cuentan para el número total de páginas.)

Orientación Editorial

A continuación le presentamos una serie de elementos relacionados con el documento de entrega que permite la creación del documento con un formato estandarizado y al mismo tiempo que se alinea a la plataforma OJS.

Figura 1. (Negritas)

Ciclo administrativo (Cursiva)



Nota. Martínez, 2019, Ciencia Ergo Sum. <http://www.cienciaergosum.edum.mx>