



# Departamento de Eléctrica, Electrónica y Telecomunicaciones (DEET)



## Carrera de Electrónica Y automatización

### Fundamentos de Programación

Perfil del Proyecto

Presentado por:

- Comina Fonseca Francisco Israel
- Tutillo Heredia Dylan Josue
- Alquinga Quishpe Kevin Omar

Tutor académico:

- Ruiz Robalino Jenny Alexandra

Ciudad: Quito

Fecha: 20/11/2025

# **Índice**

---

## **PERFIL DE PROYECTO**

1. Introducción....
2. Planteamiento del trabajo....
  - 2.1 Formulación del problema....
  - 2.2 Justificación....
3. Sistema de Objetivos....
  - 3.1. Objetivo General.....
  - 3.2. Objetivos Específicos (03)
4. Alcance....
  - 5.1 Metodología (Marco de trabajo 5W+2H) ....
6. Ideas a Defender ....
7. Resultados Esperados
8. Viabilidad(Ej.) .....
  - 8.1 Humana....
    - 8.1.1 Tutor Empresarial ....
    - 8.1.2 Tutor Académico....
    - 8.1.3 Estudiantes....
  - 8.2 Tecnológica....
    - 8.2.1 Hardware....
    - 8.2.2 Software....
9. Cronograma: ....
10. Bibliografía....

# **1. Introducción**

En la actualidad, los pequeños negocios de distribución de alimentos continúan enfrentando dificultades en la gestión de sus procesos internos especialmente cuando dependen de métodos manuales para el registro, control de inventarios. La ausencia de sistemas automatizados influye en la eficiencia operativa y en la precisión del manejo de datos. Esto es visible en microempresas como por ejemplo las distribuidoras de huevos, donde el volumen de transacciones diarias exige un control ordenado y confiable. Como señala Laudon y Laudon (2020), “la información es un recurso clave para la toma de decisiones y las organizaciones que no administran adecuadamente estos datos se exponen a errores, pérdidas económicas y descoordinación operativa” (p. 37). Este planteamiento hace énfasis en la importancia de poner soluciones como la tecnología en negocios pequeños.

El control de inventario es un proceso básico en cualquier distribuidora que maneje productos físicos, ya que permite mantener niveles adecuados de existencias, reducir pérdidas y garantizar continuidad operativa. En una distribuidora de huevos, donde el producto es perecedero y la demanda puede ser variable y llevar un control preciso de entradas y salidas es importante para evitar tanto el desabastecimiento como el exceso de mercancía. Según Heizer y Render (2015), “el inventario representa uno de los recursos más costosos dentro de una empresa, por lo que administrarlo adecuadamente es fundamental para asegurar la eficiencia operativa” (p. 62).

El manejo manual del inventario provoca errores de registro, datos incompletos y desactualización del stock, lo que afecta negativamente la planificación y el abastecimiento. De acuerdo con Ballou (2004), una gestión de inventario deficiente puede generar costos innecesarios, pérdidas de mercancía y fallas en la atención de la demanda, lo que demuestra la importancia de contar con métodos sistematizados que permitan evaluar las existencias de forma precisa.

Debido a lo anterior, este proyecto se enfoca en el desarrollo de un sistema básico de control de inventario para una distribuidora de huevos. El software permitirá registrar entradas y salidas del producto, visualizar el stock disponible y mantener un registro actualizado de las existencias. A través de la programación estructurada, el sistema busca reducir errores de conteo, mejorar el seguimiento de los recursos y optimizar el manejo de inventario, fortaleciendo así la organización interna del negocio y contribuyendo a un mejor uso de sus recursos.

## **2. Planteamiento del trabajo**

### **2.1 Formulación del problema**

El control manual del inventario en una distribuidora de huevos genera errores frecuentes en el registro de existencias, desactualización del stock y dificultad para conocer la cantidad real de producto disponible. Estos problemas afectan la planificación diaria y aumentan el riesgo de pérdidas o desabastecimiento. Por ello, surge la necesidad de desarrollar un sistema que permita registrar de manera automática las entradas y salidas de huevos. Según Laudon y Laudon (2020), la gestión manual incrementa las posibilidades de error y limita la eficiencia operativa, lo que coincide con la situación encontrada en este tipo de negocios.

El problema principal se puede formular de la siguiente manera:

**¿Cómo desarrollar un programa que permita controlar el inventario de una distribuidora de huevos, registrando entradas y salidas del producto para mantener un stock actualizado y reducir errores derivados del manejo manual?**

### **2.2 Justificación**

Un sistema de control de inventario es importante para organizar mejor una distribuidora de huevos. Cuando se maneja el inventario a mano, es fácil cometer errores, perder información y no saber exactamente cuántos productos hay, lo que complica las decisiones diarias, como cuántos productos pedir o distribuir.

Con un programa, se puede llevar un control más preciso y rápido de lo que entra y sale del inventario, reduciendo errores y asegurando que siempre esté actualizado. En este proyecto se ofrece una solución simple y efectiva usando herramientas fáciles como Codeblocks, ayudando a mejorar la gestión del inventario sin complicaciones y en poco tiempo.

### **3. Sistema de Objetivos**

#### **3.1. Objetivo General**

Desarrollar un programa de control de datos y gestión para una distribuidora de huevos, utilizando lenguaje de programación aprendido en clases en este caso leguaje C con el fin de aplicar lo aprendido para optimizar el control del inventario, mejorar la organización administrativa del local de ventas y reducir los errores en los procesos de venta y facturación.

#### **3.2. Objetivos Específicos**

- Definir las funciones que debe realizar el sistema, como registrar entradas, registrar salidas y mostrar el stock disponible, aplicando técnicas básicas como listas de requerimientos y diagramas simples para asegurar que el sistema cumpla con las funciones esenciales que permitan controlar las existencias correctamente y reducir errores del manejo manual.
- Determinar características como rapidez del programa, estabilidad, interfaz sencilla y claridad en las opciones del menú aplicando principios de desarrollo básico vistos en clase, como modularidad, validación de datos, eficiencia en estructuras de control y uso adecuado de CodeBlocks para garantizar que el sistema sea fácil de usar, confiable y funcional.
- Construir un programa que permita administrar el inventario mediante el uso de variables, funciones, arreglos y ciclos utilizando CodeBlocks como entorno de desarrollo, el lenguaje C como base, y aplicando estructuras condicionales y repetitivas ya aprendidas para obtener un sistema funcional y académico que permita automatizar el control de inventario y demostrar el dominio de herramientas y técnicas vistas en clase.

## **4. Alcance**

El proyecto se limita al diseño y desarrollo de un sistema básico de control de inventario para una distribuidora de huevos. El sistema permitirá registrar entradas y salidas del producto y mostrar en pantalla el stock actualizado.

Para el desarrollo del software se utilizará el entorno de programación CodeBlocks, aplicando programación estructurada en lenguaje C, así como conceptos fundamentales las cuales son estructuras secuenciales, condicionales (if, switch), ciclos repetitivos (while, do-while, for), arreglos y manejo básico de entrada y salida de datos. Estas herramientas permitirán construir un programa simple, claro y funcional.

El alcance del proyecto no incluye módulos adicionales como reportes avanzados, bases de datos, interfaz gráfica, conexión con sistemas externos o funciones comerciales. El sistema operará localmente en la consola del computador y estará orientado exclusivamente a resolver el problema del control interno del inventario.

## **5. Marco Teórico**

### **Entornos de Desarrollo (IDEs) utilizados.**

Para el desarrollo del programa se empleó un Entorno de Desarrollo Integrado (IDE), que facilita la escritura, compilación, depuración y ejecución del código. Estos entornos hacen que el proceso de programación sea ordenado y seguro al automatizar tareas básicas y proporcionar herramientas visuales para el programador.

#### **CodeBlocks**

El IDE principal usado es CodeBlocks el cual es una herramienta muy recomendada para estudiantes que programan en C porque es fácil de usar, tiene una interfaz sencilla y funciona bien con distintos compiladores. CodeBlocks incluye:

- Un editor que resalta el código.
- Un sistema de compilación que usa GCC (MinGW).
- Un depurador integrado para encontrar errores mientras el programa corre.
- La opción de ejecutar el programa directamente, sin usar otras herramientas.

Gracias a su simplicidad, podemos enfocarnos en aprender la estructura del lenguaje C y la lógica del programa, sin complicarnos con configuraciones avanzadas.

### **Lenguaje de programación C**

El sistema está hecho en lenguaje C, uno de los lenguajes más porque es rápido, eficiente y tiene una estructura clara. C es muy útil ya que es la base de muchos lenguajes usados en sistemas y en microcontroladores.

Características relevantes para el proyecto:

- La programación estructurada permite dividir el programa en funciones, lo que ayuda a mantener el código ordenado.
- El manejo simple de variables es útil para controlar inventarios, cantidades y precios.

- Los condicionales y los ciclos forman la base del menú interactivo y de las tareas repetitivas del sistema.
- Es un lenguaje que permite que el programa funcione bien incluso en computadoras con pocos recursos.

Conceptos fundamentales de programación utilizados

El sistema de gestión implementado se apoya en varios conceptos esenciales de la programación, indispensables para el correcto funcionamiento del software.

### **Variables y tipos de datos**

Se usan para guardar información del negocio, como cantidades de huevos, datos de clientes, totales de facturación y ventas, utilizando números y cadenas de texto según lo que se necesite almacenar.

### **Estructuras de control**

Incluyen condicionales y ciclos que permiten validar opciones del menú, repetir tareas y guiar el flujo del sistema según las decisiones del usuario.

### **Arreglos o estructuras de almacenamiento**

Permiten guardar listas simples de datos, como clientes o ventas, sin usar bases de datos externas, lo cual es adecuado para un proyecto básico.

## **5.1 Metodología (Marco de trabajo 5W+2H)**

La metodología utilizada se basa en el marco 5W+2H, que permite planificar y organizar el proyecto para que se vea mejor respondiendo preguntas sobre qué se hará, cómo se hará, quién participará, cuándo se realizará, por qué es necesario, cuánto costará y qué porcentaje del trabajo se ha completado. Con esto es más sencillo la organización de las actividades, la asignación de prioridades y asegura que todas las fases del programa, como análisis, diseño, programación y pruebas, se lleven a cabo correctamente.

|         |   |
|---------|---|
| ¿QUÉ?   | Desarrollar un sistema básico de control de inventario para una distribuidora de huevos, centrado en registrar entradas, salidas y visualizar el stock disponible, identificando los requisitos funcionales y no funcionales necesarios para su funcionamiento. |
| ¿CÓMO?  | Aplicando conceptos fundamentales de programación: variables, funciones, estructuras de control, arreglos y validación de datos. El desarrollo se realiza en el IDE CodeBlocks, siguiendo un diseño modular que divide el programa en procesos independientes.  |
| ¿QUÍEN? | El trabajo es realizado por el grupo 1, bajo la supervisión del tutor académico y con apoyo informativo del tutor empresarial.  |

|                          |   |
|--------------------------|---|
| <b>¿CUÁNDO?</b>          | El proyecto se desarrolla durante el período académico establecido, siguiendo un cronograma que incluye fases de análisis, programación, pruebas y correcciones.  |
| <b>¿POR QUÉ?</b>         | Porque los procesos manuales de la distribuidora generan errores frecuentes, pérdida de información y desorganización. El programa automatiza tareas, mejora la eficiencia operativa y permite que apliquemos nuestros conocimientos en un caso real. |
| <b>¿CUÁNTO?</b>          | El proyecto requiere recursos tecnológicos básicos: una computadora con un IDE para C, almacenamiento local y tiempo de trabajo. El costo económico es bajo, ya que se utilizan herramientas gratuitas y hardware comúnmente disponible.              |
| <b>% DE CUMPLIMIENTO</b> | Este campo permite registrar el avance del proyecto en cada etapa, de acuerdo con el cronograma definido. Se completa conforme el proyecto avanza.  |

## 6. Ideas a Defender

Este proyecto se fundamenta en varias ideas clave que justifican su desarrollo y orientan su diseño hacia un sistema funcional, sencillo y adecuado. Las ideas a defender se centran en la importancia del control de inventario y en el uso adecuado de herramientas aprendidas en clases.

- **Automatizar el inventario reduce errores.**

Un sistema digital evita fallas comunes del conteo manual y permite llevar un registro más preciso de las entradas y salidas del producto.

- **Un sistema simple puede resolver un problema real.**

Aunque sea básico, un programa hecho con programación estructurada puede mejorar significativamente el control del inventario.

- **Un control de inventario preciso mejora la toma de decisiones.**

Un stock actualizado ayuda a planificar correctamente compras, distribución y uso de recursos dentro del negocio.

## 7. Resultados Esperados

El desarrollo del programa está orientado a obtener resultados concretos que aporten valor tanto al negocio como a nuestro proceso formativo. Entre los resultados esperados se incluyen:

### **Un sistema funcional para registrar entradas y salidas del inventario.**

Se espera obtener un programa capaz de registrar correctamente cada movimiento del inventario, ya sea ingreso o salida de producto, y actualizar inmediatamente el stock disponible.

### **Implementación correcta de conceptos de programación estructurada.**

El sistema debe reflejar el uso adecuado de herramientas como variables, condicionales, ciclos, arreglos, funciones y modularidad, integrando todos estos elementos en un programa claro y organizado.

**Uso adecuado del entorno CodeBlocks para compilar y ejecutar el sistema.**

Se espera manejar correctamente el IDE CodeBlocks, desde la creación del proyecto hasta la compilación y prueba del programa final, demostrando dominio básico del entorno de desarrollo.

**Reducción de errores comunes en el control manual del inventario.**

El sistema permitirá minimizar errores de conteo, pérdidas de datos o inconsistencias, ofreciendo un registro más confiable y accesible para las operaciones internas de la distribuidora.

**Un programa simple, práctico y adecuado para el nivel académico.**

Se espera obtener un resultado acorde con lo que un estudiante de primer semestre puede desarrollar: funcional, organizado y con potencial de mejoras futuras.

## 8. Viabilidad

| Cantidad       | Descripción  | Valor Unitario (USD) | Valor Total (USD) |
|----------------|--|----------------------|-------------------|
| Equipo en casa |  |                      |                   |
| 1              | Laptop (mínimo: 8GB RAM, 256GB SSD, procesador Ryzen 5 o i5 equivalente) | 600                  | 600               |
| Software       |  |                      |                   |
| 1              | Sistema Operativo Windows 10 u 11  | 0*                   | 0*                |
| 1              | CodeBlocks (IDE para C)  | 0                    | 0                 |
| 1              | Visual Studio Code (opcional)  | 0                    | 0                 |
| 1              | Compilador MinGW / GCC   | 0                    | 0                 |
| <b>TOTAL</b>   |  |                      | <b>600</b>        |

### 8.1 Humana

#### 8.1.1 Tutor Empresarial

Ing. ...

- Responsabilidades**

Proporcionar información sobre cómo funciona la distribuidora, definir los requisitos del sistema y asegurarse de que el software cumpla con las necesidades del negocio.

### **8.1.2 Tutor Académico**

**Ing. Jenny Ruiz**

- Responsabilidades**

Supervisar y orientar el desarrollo del programa, asegurando que el diseño, el código y la documentación sean correctos, y ayudando al estudiante a usar adecuadamente las herramientas y el IDE.

### **8.1.3 Estudiantes**

- Responsabilidades**

Investigar y analizar los requisitos del proyecto, diseñar la estructura del sistema y programarlo en C, realizar pruebas y corregir errores, documentar el trabajo y entregar los avances y la versión final del programa.

## **8.2 Tecnológica**

### **8.2.1 Hardware**

|                       | <b>Requisitos mínimos</b>                 | <b>Disponibilidad</b> |
|-----------------------|---|-----------------------|
| <b>Memoria RAM</b>    | 4 GB de RAM                               | <b>Alta</b>           |
| <b>Almacenamiento</b> | <b>10 GB de espacio de almacenamiento</b> | <b>Alta</b>           |

### **8.2.2 Software**

|                          | <b>Requisitos mínimos</b>   | <b>Disponibilidad</b> |
|--------------------------|---|-----------------------|
| <b>Sistema Operativo</b> | Se recomienda Windows 10 u 11, macOS 10.10 o Ubuntu 16  | <b>Alta</b>           |
| <b>IDE</b>               | <b>Es recomendable Visual Studio Code debido a su conexión con FTP, sin embargo, cualquier IDE con esta funcionalidad funciona.</b> | <b>Alta</b>           |

## **9. Conclusiones y recomendaciones**

### **9.1 Conclusiones**

### **9.2 Recomendaciones**

## 10. Planificación para el Cronograma:

Debe insertar una imagen clara y legible de la planificación del proyecto a desarrollar.

| # | TAREA  | INICIO     | FIN        |
|---|--|------------|------------|
| 1 | Identificación y análisis del problema   | 4/11/2025  | 6/11/2025  |
| 2 | Definición de los objetivos  | 7/11/2025  | 11/11/2025 |
| 3 | Investigación teórica (conceptos, herramientas de programación, métodos de registro y control) | 12/11/2025 | 18/11/2025 |
| 4 | Documentación (primer avance)  | 19/11/2025 | 23/11/2025 |
| 5 | Diseño del sistema y desarrollo del código   | 24/11/2025 | 29/11/2025 |
| 6 | Pruebas del sistema y corrección de errores  | 30/11/2025 | 2/12/2025  |
| 7 | Fin del informe  | 3/12/2025  | 9/12/2025  |
| 8 | Exposición del informe   | 10/12/2025 | 10/12/2025 |

## 11. Referencias

Laudon, K. C., & Laudon, J. P. (2020). *Management Information Systems: Managing the Digital Firm* (16.<sup>a</sup> ed.). Pearson.

O'Brien, J. A., & Marakas, G. M. (2017). *Management Information Systems* (10.<sup>a</sup> ed.). McGraw-Hill Education.

Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach* (8.<sup>a</sup> ed.). McGraw-Hill Education.

Pressman, R. S., & Maxim, B. (2015). *Software Engineering: A Practitioner's Approach* (9.<sup>a</sup> ed.). McGraw-Hill Education.

Sommerville, I. (2016). *Software Engineering* (10.<sup>a</sup> ed.). Pearson.

Stair, R., & Reynolds, G. (2019). *Principles of Information Systems* (13.<sup>a</sup> ed.). Cengage Learning.

Turban, E., & Volonino, L. (2018). *Information Technology for Management: Driving Digital Transformation* (11.<sup>a</sup> ed.). Wiley.

Whitten, J. L., & Bentley, L. D. (2019). *Systems Analysis and Design Methods* (8.<sup>a</sup> ed.). McGraw-Hill Education.

## **Anexos.**

**Anexo I. Crono**

**Anexo II. Historia de Usuario**