Manual Técnico del Software

# Portada

Título del documento: Manual Técnico del Software

Nombre del software: GA10-220501097-AA10-EV01

Versión del software: 1.0

Nombre del programa de formación: Tecnólogo en Análisis y Desarrollo de Software

Ficha de formación: 220501097

Fecha de entrega: Julio 2025

Nombres completos de los aprendices/desarrolladores: [Escribir nombres]

Nombre del instructor: [Escribir nombre]

Centro de formación y regional del SENA: [Ejemplo: Centro de Comercio y Servicios - Regional Valle]

# 1. Introducción

El propósito de este documento es documentar el funcionamiento técnico del software   
El software fue desarrollado como parte de una actividad de aprendizaje del programa Tecnólogo en Análisis y Desarrollo de Software.   
Este sistema está diseñado para facilitar la gestión de información relacionada con [describir funcionalidad principal], dirigido a [tipo de usuario].   
El desarrollo se realizó durante la etapa productiva, aplicando buenas prácticas de programación y diseño.

# 2. Prerrequisitos de instalación

a. Requisitos de hardware:

- Procesador mínimo: Intel i3

- RAM recomendada: 4 GB

- Espacio en disco: 1 GB

- Resolución mínima de pantalla: 1366x768

b. Requisitos de software:

- Sistema operativo: Windows 10 / Linux / macOS

- Node.js v18 o superior

- MongoDB o MySQL instalado

- Navegador web (Chrome recomendado)

- Editor de código: Visual Studio Code

- Git y dependencias necesarias instaladas

# 3. Frameworks y estándares utilizados

- Frontend: React.js, Bootstrap

- Backend: Node.js, Express

- Base de datos: MongoDB / MySQL

Estándares seguidos:

- RESTful API

- Arquitectura MVC

- Estilo de código Airbnb (JavaScript)

- Buenas prácticas de seguridad y validación de datos

# 4. Diagrama de casos de uso

El sistema permite que los actores realicen diversas acciones. A continuación se describe un ejemplo:

Actor: Administrador → Casos de uso: Registrar usuario, Ver reportes, Eliminar producto

[Insertar imagen de diagrama de casos de uso aquí]

# 5. Modelo entidad-relación (MER)

Representa gráficamente las entidades del sistema, sus atributos y relaciones entre ellas.

[Insertar imagen del modelo entidad-relación aquí]

Ejemplo de entidades: Usuarios, Productos, Pedidos.

# 6. Diccionario de datos

Tabla: usuarios  
- id\_usuario: INT, Identificador único  
- nombre: VARCHAR(100), Nombre del usuario  
- correo: VARCHAR(100), Correo electrónico  
- contraseña: VARCHAR(255), Contraseña encriptada  
  
Tabla: productos  
- id\_producto: INT, Identificador único  
- nombre: VARCHAR(100), Nombre del producto  
- precio: DECIMAL, Precio del producto  
- stock: INT, Cantidad disponible

# 7. Scripts de instalación

# Paso 1: Clonar el repositorio  
git clone https://github.com/usuario/proyecto.git  
  
# Paso 2: Instalar dependencias del backend  
cd backend  
npm install  
node server.js  
  
# Paso 3: Instalar dependencias del frontend  
cd frontend  
npm install  
npm start

También se debe configurar la base de datos, crear un usuario administrador inicial y definir variables de entorno en un archivo `.env`.

# 8. Diagrama de componentes

Representa la estructura general del sistema, incluyendo:

- Frontend (componentes de React)

- Backend (controladores, rutas, modelos)

- Base de datos y conexiones entre los módulos

[Insertar imagen del diagrama de componentes aquí]

# 9. Bibliografía

- Documentación de React: https://reactjs.org  
- Node.js: https://nodejs.org  
- Express.js: https://expressjs.com  
- MongoDB: https://www.mongodb.com/docs  
- RESTful API: https://restfulapi.net  
- Stack Overflow, MDN Web Docs, GitHub