# Documento de Diseño y Plan de Pruebas

## 1. Introducción

Este documento describe el diseño del **Sistema de Procesamiento de Imágenes**, detallando los **Tipos Abstractos de Datos (TADs)** utilizados, las operaciones implementadas y su estructura. También se incluye un **plan de pruebas** para validar el funcionamiento del comando proyeccion2D.

## 2. Descripción del Sistema

El sistema implementado permite la manipulación y procesamiento de imágenes en formato **PGM (P2)**, con funcionalidades como:

* Carga de imágenes individuales (cargar\_imagen).
* Carga de volúmenes de imágenes (cargar\_volumen).
* Consulta de información (info\_imagen y info\_volumen).
* Generación de proyecciones en 2D (proyeccion2D).

El código está desarrollado en **C++**, utilizando estructuras de datos como **vectores** (std::vector) para almacenar las imágenes y el volumen de datos.

## 3. Tipos Abstractos de Datos (TADs)

### ****TAD: Imagen****

**Descripción:** Representa una imagen en formato **PGM** cargada en memoria.

**Datos:**

* imageData: Matriz de enteros que almacena los valores de los píxeles.
* width: Ancho de la imagen.
* height: Alto de la imagen.
* maxPixelValue: Valor máximo de píxel permitido.
* imageFilename: Nombre del archivo cargado.

**Operaciones:**

* loadImage(nombre\_archivo): Carga una imagen en memoria.
* infoImage(): Muestra información de la imagen cargada.

### ****TAD: Volumen de Imágenes****

**Descripción:** Representa un conjunto de imágenes de la misma dimensión cargadas como un volumen tridimensional.

**Datos:**

* volumeData: Vector 3D que almacena todas las imágenes del volumen.
* volume: Nombre base del volumen cargado.
* width, height: Dimensiones de las imágenes.
* maxPixelValue: Valor máximo de píxel.

**Operaciones:**

* loadVolume(base\_nombre, n\_im): Carga un volumen de n\_im imágenes.
* infoVolume(): Muestra información del volumen.
* proyeccion2D(direccion, criterio, archivo\_salida): Genera una proyección en 2D del volumen.

## 4. Diseño del Sistema y Funcionalidades

### ****4.1. Estructura General****

El sistema sigue un modelo **basado en comandos**, donde el usuario puede interactuar mediante la terminal.

### ****4.2. Entradas y Salidas****

| **Función** | **Entrada** | **Salida** |
| --- | --- | --- |
| cargar\_imagen | Nombre de un archivo .pgm | Imagen cargada en memoria. |
| cargar\_volumen | Base del nombre y número de imágenes | Volumen de imágenes cargado. |
| info\_imagen | Ninguna | Información de la imagen cargada. |
| info\_volumen | Ninguna | Información del volumen cargado. |
| proyeccion2D | Dirección (x, y, z), criterio, salida | Archivo PGM con la proyección 2D. |

## 5. Plan de Pruebas para proyeccion2D

### ****5.1. Casos de Prueba****

| **ID** | **Descripción** | **Entrada** | **Salida Esperada** |
| --- | --- | --- | --- |
| P1 | Proyección en el eje Z (máximo) | proyeccion2D z max salida.pgm | Imagen con valores máximos en Z. |
| P2 | Proyección en el eje Z (mínimo) | proyeccion2D z min salida.pgm | Imagen con valores mínimos en Z. |
| P3 | Proyección en el eje Z (promedio) | proyeccion2D z prom salida.pgm | Imagen con valores promedio en Z. |
| P4 | Proyección en el eje Z (mediana) | proyeccion2D z med salida.pgm | Imagen con la mediana de valores en Z. |
| P5 | Proyección en el eje Y (máximo) | proyeccion2D y max salida.pgm | Imagen con valores máximos en Y. |
| P6 | Proyección en el eje Y (mínimo) | proyeccion2D y min salida.pgm | Imagen con valores mínimos en Y. |
| P7 | Proyección en el eje Y (promedio) | proyeccion2D y prom salida.pgm | Imagen con valores promedio en Y. |
| P8 | Proyección en el eje Y (mediana) | proyeccion2D y med salida.pgm | Imagen con la mediana de valores en Y. |
| P9 | Proyección en el eje X (máximo) | proyeccion2D x max salida.pgm | Imagen con valores máximos en X. |
| P10 | Proyección en el eje X (mínimo) | proyeccion2D x min salida.pgm | Imagen con valores mínimos en X. |
| P11 | Proyección en el eje X (promedio) | proyeccion2D x prom salida.pgm | Imagen con valores promedio en X. |
| P12 | Proyección en el eje X (mediana) | proyeccion2D x med salida.pgm | Imagen con la mediana de valores en X. |
| P13 | Proyección sin volumen cargado | proyeccion2D x max salida.pgm | Mensaje de error: No hay volumen cargado. |

### ****5.2. Criterios de Aceptación****

* La proyección debe generarse correctamente según el criterio seleccionado.
* El tamaño de la imagen proyectada debe coincidir con la resolución esperada.
* Si no hay volumen cargado, el sistema debe mostrar un mensaje de error.

**Guía para Clonar, Compilar y Ejecutar el Programa desde GitHub**

**1. Clonar el Repositorio**

Para obtener el código fuente del programa, debe clonar el repositorio desde GitHub.

**🔹 Instrucciones:**

1. **Abrir la terminal (Linux/macOS) o Git Bash (Windows).**
2. **Ejecutar el siguiente comando:**
3. git clone https://github.com/jeisonAlfonso/entrega1-ED.git
4. **Entrar a la carpeta del repositorio clonado:**
5. cd entrega1-ED

**2. Compilar el Programa**

Una vez dentro de la carpeta del repositorio, debe compilar el código.

**🔹 En Linux/macOS:**

g++ -o programa main.cpp

**🔹 En Windows (con MinGW):**

g++ -o programa.exe main.cpp

Si la compilación es exitosa, se generará un archivo ejecutable llamado programa (en Linux/macOS) o programa.exe (en Windows).

**3. Ejecutar el Programa**

Después de compilar, puede ejecutar el programa con los siguientes comandos:

**🔹 En Linux/macOS:**

./programa

**🔹 En Windows:**

programa.exe

Esto iniciará el sistema de procesamiento de imágenes en la terminal.

**4. Uso de los Comandos en el Programa**

Para utilizar el programa, siga las instrucciones que se muestran en la terminal después de ejecutar el comando de salida.