

## Proyecto de Estructura de Datos: Entrega 1

Esteban Navas

Juan Bello

Santiago Hernández

Pontifica Universidad Javeriana

## Resumen

En el presente artículo se presentarán las pautas de ingeniería implementadas para realizar el desarrollo del componente 1 del proyecto de Estructuras de Datos el cual se desarrollará durante el semestre actual 2025-1, describiendo el desarrollo utilizado para el funcionamiento de las entradas, salidas y condiciones para el procedimiento principales y las operaciones auxiliares, basándonos en los TADs definidos para su desarrollo y la representación gráfica de las operaciones principales.

## Descripción del Problema

Teniendo en cuenta el objetivo general del proyecto de construir un sistema para la construcción de imágenes utilizando conceptos de Estructura de Datos y la modulación de los componentes para sus diferentes funciones se empezara la construcción del sistema primero mediante la construcción del primer componente a cargo de la Proyección de Imágenes donde se buscara generar una representación planar a partir de un objeto o escena 3D a través de los comandos ya definidos con los cuales se debe implementar el componente.

## Procedimiento Principal y Operaciones Auxiliares

Descripción:

### 1. Cargar\_imagen

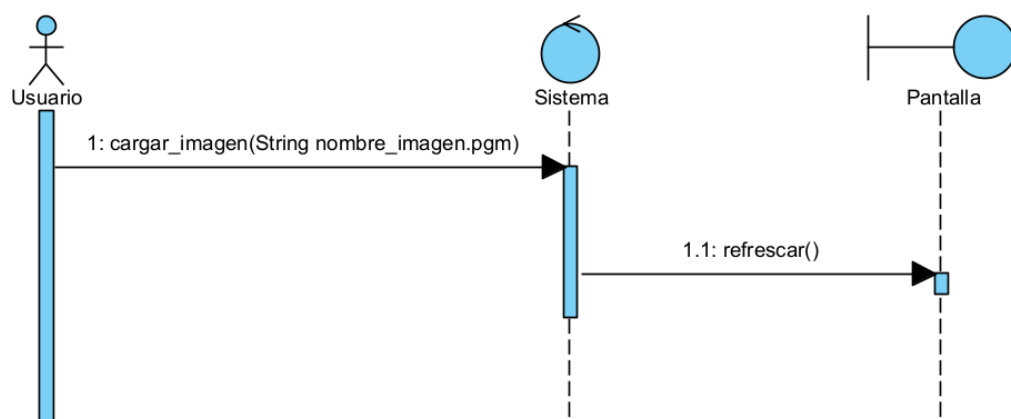


Figura 1: Procedimiento de Cargar\_imagen

- Procedimiento

Como se ve en la Figura 1, el usuario introduce el comando brindando el nombre de la imagen a cargar en el sistema, el sistema refresca la pantalla con el mensaje de éxito o error

- Entradas

Imágen de extensión pgm

- Salidas

La imagen se carga en memoria y dependiendo de el resultado de la operación se imprime alguno de los dos mensajes de notificación:

- (Proceso satisfactorio) La imagen nombre\_imagen.pgm ha sido cargada.
- (Mensaje de error) La imagen nombre\_imagen.pgm no ha podido ser cargada.

- Condiciones

- Solo se podrá cargar una imagen por sesión.
- El comando debe mostrar el mensaje de Proceso satisfactorio en caso de que se haya podido cargar la imagen en memoria o el Mensaje de error en caso de que haya ocurrido un problema

## 2. Cargar\_volumen

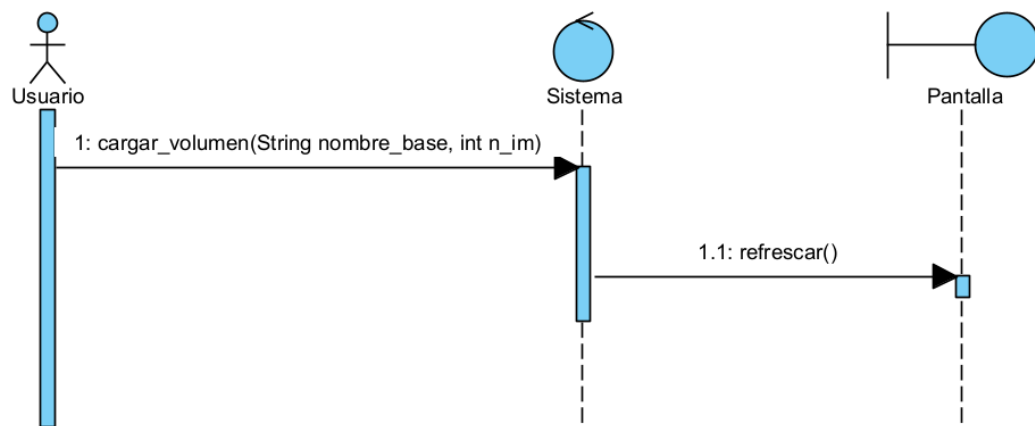


Figura 2: Procedimiento de Cargar\_volumen

- Procedimiento

- Como se ve en la Figura 2, el usuario introduce el comando brindando el nombre base de la serie de imágenes a cargar en el sistema, el sistema refresca la pantalla con el mensaje de éxito o error

- Entradas

Nombre de una serie ordenada de imágenes

Numero de imágenes de la serie ordenada

- Salidas

La serie ordenada de imágenes se carga en memoria y dependiendo de el resultado de la operación se imprime alguno de los dos mensajes de notificación:

- (Proceso satisfactorio) El volumen nombre\_base ha sido cargado.
- (Mensaje de error) El volumen nombre\_base no ha podido ser cargado

- Condiciones

- Solo se podrá cargar una imagen por sesión.
- La serie podrá tener como máximo 99 imágenes
- Todas las imágenes de la serie deben estar nombradas nombre\_base xx.pgm, donde xx corresponde a dos dígitos de identificación de la posición de la imagen en la serie
- El comando debe mostrar el mensaje de Proceso satisfactorio en caso de que se haya podido cargar la serie de imágenes en memoria o el Mensaje de error en caso de que haya ocurrido un problema

### 3. Info\_imagen

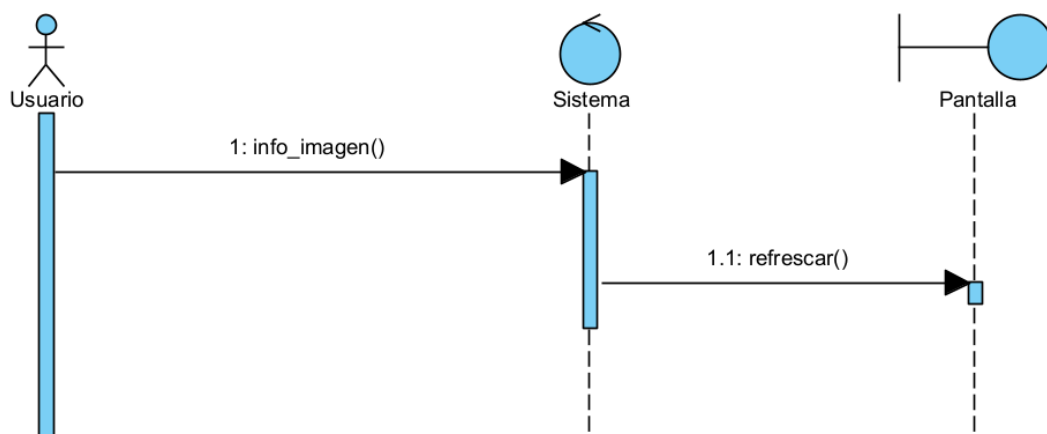


Figura 3: Procedimiento de Info\_imagen

- Procedimiento
- Como se ve en la Figura 3, el usuario introduce el comando , el sistema busca si hay una imagen en memoria y refresca la pantalla con el mensaje de éxito o error
- Entradas  
No se necesita de entradas por parte del usuario
- Salidas  
Se muestra la información de la imagen cargada actualmente en memoria
  - (Proceso satisfactorio) Imagen cargada en memoria:  
nombre\_imagen.pgm , ancho: W , alto: H .
  - (Mensaje de error) No hay una imagen cargada en memoria
- Condiciones  
Si no se ha cargado aún una imagen en memoria, el comando debe mostrar el mensaje de error.

#### 4. Info\_Volumen

- Procedimiento

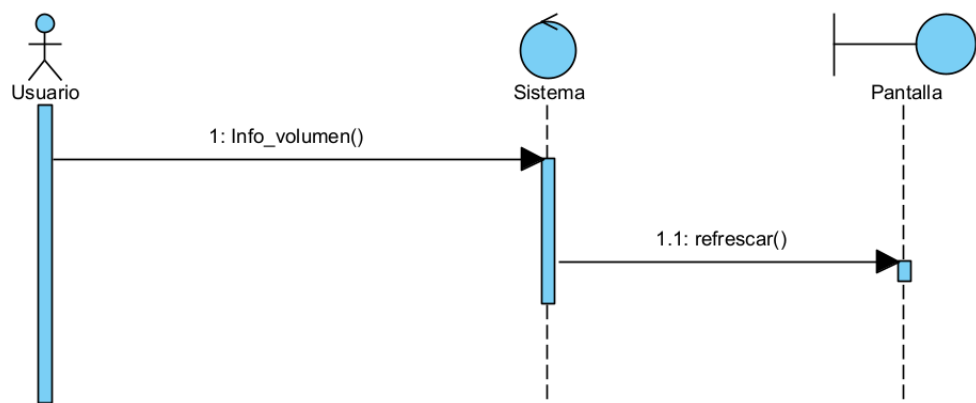


Figura 4: Procedimiento de Info\_volumen

- Procedimiento
  - Como se ve en la Figura 4, el usuario introduce el comando , el sistema busca si hay un volumen en memoria y refresca la pantalla con el mensaje de éxito o error
- Entradas
 

No se necesita de entradas por parte del usuario
- Salidas
 

Se muestra la información del volumen cargado actualmente en memoria

  - (Proceso satisfactorio) Volumen cargado en memoria: nombre\_base , tamaño: n\_im , ancho: W , alto: H .
  - (Mensaje de error) No hay un volumen cargado en memoria.
- Condiciones
 

Si no se ha cargado aún una imagen en memoria, el comando debe mostrar el mensaje de error.

## 5. Proyeccion2D

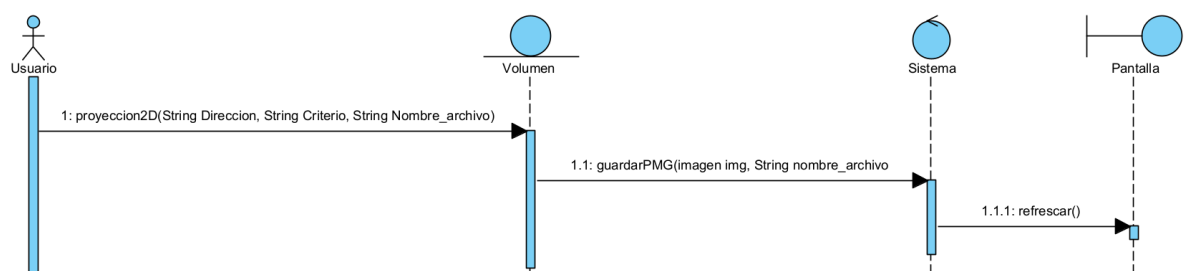


Figura 5: Procedimiento de Proyeccion2D

- Procedimiento
 

Como se ve en la Figura 5, el usuario introduce el comando con la Direccion, criterio y nombre del archivo para la creación de la proyección, se produce la proyección que se trata como imagen y se guarda como

archivo con la función guardarPMG, el sistema refresca la pantalla para imprimir los mensajes de confirmacion

- Entradas

Dirección a la que se desea que se recorra cada posición en el plano

Criterio para la generación de la proyección

Nombre de imagen donde guardar la Proyeccion2D

- Salidas

Se debe generar la proyección 2D y guardarse en un archivo con nombre ya dado por el usuario además se debe notificar del resultado de la operación con uno de los dos mensajes:

- (Proceso satisfactorio) La proyección 2D del volumen en memoria ha sido generada y almacenada en el archivo nombre\_archivo.pgm .
- (Mensajes de error) El volumen aún no ha sido cargado en memoria.

- Condiciones

- La dirección puede ser una entre x (en dirección de las columnas), y (en dirección de las filas) o z (en dirección de la profundidad).
- El criterio puede ser uno entre minimo (el valor mínimo de intensidad), maximo (el valor máximo de intensidad), promedio (el valor promedio de intensidad) o mediana (el valor mediano de intensidad)
- Una vez generada la proyección, debe guardarse como imagen en formato PGM como nombre\_archivo.pgm .
- El comando solo puede funcionar sobre volúmenes



## 6. guardarPMG

- Procedimiento

Como se ve en la Figura 5, la función se encarga de generar el archivo con la proyección realizada

- Entradas

Imagen con la proyección generada

Nombre del archivo a generar

- Salidas

Archivo con extensión .pmg con la proyección generada

- Condiciones

Se verifica que al momento de llamar `proyeccion2D` exista la imagen con la proyección

## Desarrollo de TADs

Definición de TADs:

TAD imagen:

- Datos mínimos:
  - código : Cadena de caracteres utilizada para definir que comando se empleara sobre la imagen
  - W: entero con el ancho de la imagen
  - H: entero con la altura de la imagen
  - M: entero con el tamaño máximo de la imagen
  - val : Vector con los valores de la imagen
- Operaciones:

- guardarPGM(img, nombre\_archivo) : procesa la imagen en memoria y guarda la proyección en un archivo pgm

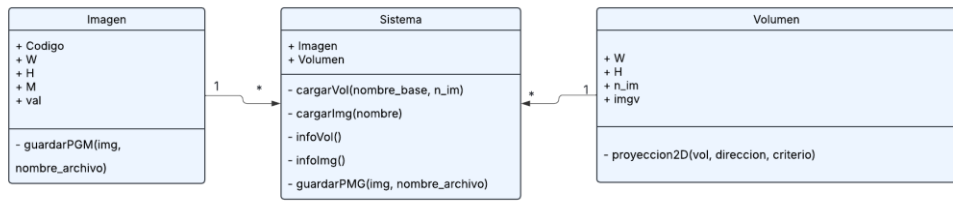
#### TAD volumen

- Datos Minimos:
  - W: Entero con el ancho de la imagen mas ancha
  - H: Entero con la altura de la imagen mas alta
  - n\_Im: Entero con el número de imágenes en el volumen
  - imgv: Vector con las imágenes que conforman el volumen
- Operaciones:
  - proyeccion2D(vol, direccion, criterio) : procesa el volumen en memoria y guarda la proyección en un archivo pgm

#### TAD sistema

- Datos mínimos:
  - Imagen: Clase de imagen a cargar en memoria
  - Volumen: Clase de volumen a cargar en memoria
- Operaciones:
  - cargarVol(nombre\_base, n\_im) : carga el volumen en el sistema
  - cargarImg(nombre) : carga la imagen en el sistema
  - infoVol() : brinda información del volumen en sistema
  - infoImg () : brinda información de la imagen en sistema
  - guardarPGM( img, nombre\_archivo)

#### Diagrama de relación



## Conclusiones

- Se debe manejar una imagen o volumen a la vez en el sistema
- Se decidió utilizar una función adicional para guardar la proyección de las imágenes