



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN  
IIC2026 — VISUALIZACIÓN DE INFORMACIÓN

## Entrega 2

**Tema:** Codificaciones visuales y utilidades de D3

**Tiempo estimado:** 6 horas

Fecha de entrega: **8 de octubre 2020** a las **20:00 hrs**

### Contexto

En el contexto de Visualización de Información, la definición de **glifo**<sup>1</sup> no está completamente en consenso. Tamara Munzner los describe<sup>2</sup> como un objeto con estructura interna, que es el producto de múltiples marcas elementales, donde mediante canales visuales se codifican datos. En esta entrega te propondremos la creación e implementación de un glifo a partir de datos utilizando la librería [D3.js](#).

### Dataset

El *dataset* que utilizarás en esta entrega corresponde al [Iris Flower Dataset](#) en formato JSON (el cual pueden encontrar en el siguiente enlace de [Canvas](#)). Este conjunto de datos contiene información taxonómica sobre varias muestras de flores del género *Iris*, que se subdivide en tres tipos: *Iris versicolor*, *Iris virginica* e *Iris setosa*. La Figura 1 muestra imágenes<sup>3</sup> de muestras de cada especie.



Figura 1: Ejemplo de las tres variaciones del género *Iris*.

Este *dataset* es altamente conocido en el mundo de **Machine Learning y Estadística**, ya que corresponde a uno de los *datasets* más simples para introducir al aprendizaje supervisado y la regresión logística multidimensional. Si no conoces ninguno de los conceptos anteriores no te preocupes, ya que en esta entrega lo utilizaremos con un objetivo completamente distinto.

<sup>1</sup>Revisamos un ejemplo de glifo en una [cápsula](#) del curso.

<sup>2</sup>En su libro: Visualization, Analysis and Design.

<sup>3</sup>[Fuente de imagen](#).

Cada una de las muestras tomadas es representada en el archivo JSON por un ítem. Cada uno tiene el siguiente formato, presentando cinco atributos distintos:

```
1  [  
2    ...  
3    {  
4      "sepalLength": 5.1,  
5      "sepalWidth": 3.5,  
6      "petalLength": 1.4,  
7      "petalWidth": 0.2,  
8      "species": "setosa"  
9    },  
10   ...  
11  ]
```

Dónde:

- Los atributos `sepalLength` y `sepalWidth` corresponden respectivamente al **largo** y **ancho** del **sépal**o de la muestra en centímetros. El sépalo corresponde a la pieza floral que envuelve a las otras en la primera fase de desarrollo de una flor.
- Los atributos `petalLength` y `petalWidth` corresponden respectivamente al **largo** y **ancho** del **pétalo** de la muestra en centímetros. El pétalo corresponde a la corola de la flor, y rodean sus órganos reproductivos.
- El atributo `species` corresponde al tipo de **especie** de *Iris* a la cual corresponde la muestra (setosa, virginica o versicolor).

Las flores del género *Iris* tienen tres **pétalos** y tres **sépalos**, y se intercalan entre ellos. En la siguiente imagen podrás ver a qué corresponde cada medida y las partes de una flor de la especie *iris*<sup>4</sup>:

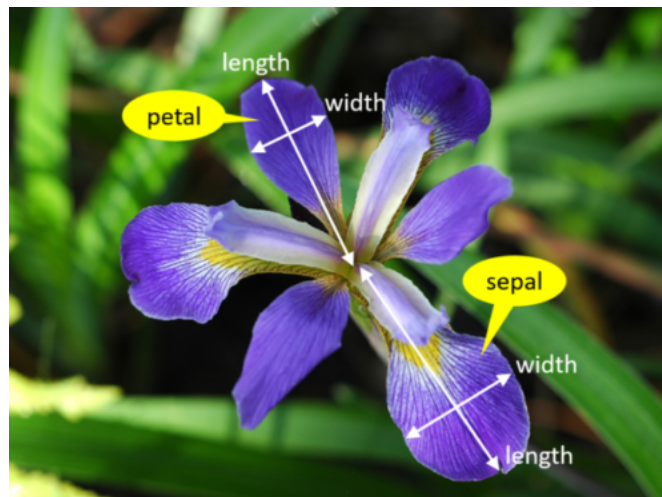


Figura 2: Partes y medidas de una flor *Iris*

<sup>4</sup>[Fuente de imagen.](#)

## 1. Representación de flores Iris

Para esta entrega deberás definir y crear, a partir de los datos entregados, un **glifo** que codifique visualmente las medidas (largos y anchos) **promedio** de las tres variedades de *Iris*. Implementarás dicha creación haciendo uso de **SVG** y **D3.js**.

### 1.1. Creación de glifo

Lo primero que tendrás que hacer es definir de cuál será la estructura de tu glifo. Es decir, de qué forma representarás visualmente cada una de las medidas de las flores: qué marcas básicas utilizarás, cómo las ordenarás espacialmente, y qué canales de dichas marcas codificarán cada atributo de medida.

Nota que **no** se pide una representación **exacta** de cada flor. Con que las representaciones estén basadas en los datos entregados y se pueda apreciar una diferencia entre especies es suficiente.

Tampoco se espera un resultado específico, la finalidad de este glifo es experimentar. Por eso, la efectividad de codificación no es necesariamente lo más importante, la creatividad también es bienvenida.

### 1.2. Implementación en D3.js

Una vez hayas definido cómo representar los datos, deberás generar tus glifos como una imagen SVG en un documento HTML. Dicho documento deberá ejecutar un programa de JavaScript que cargue los datos y genere los glifos, haciendo uso de D3.js.

El documento solo mostrará un glifo a la vez, y debe contener un elemento **dropdown** que permita cambiar el **tipo de flor** en pantalla (setosa, virginica o versicolor). Al cambiar la selección, el glifo en pantalla deberá actualizarse para representar los datos promedio del tipo elegido.

### 1.3. Consideraciones

Al momento de implementar tu solución, deberás considerar lo siguiente:

- El *dataset* contiene la información de múltiples muestras de flores por cada tipo, y el ejercicio pide visualizar el promedio de cada atributo a partir de dichas muestras. Es parte de tus responsabilidades el procesar la información para obtener dichos promedios.
- Los promedios deben ser calculados en el mismo programa que cargue y genere los glifos. No se permitirá que el archivo de datos cargado por tu programa sea diferente al entregado junto por este enunciado, o que el cálculo de los promedios sea realizado en otro lenguaje de programación de forma previa.
- Para la generación de los glifos, tu programa solo puede hacer uso de funciones nativas de JavaScript o provistas por D3.js, y no debe arrojar errores en la consola del navegador.
- Puedes hacer uso tanto de la versión 5 o 6 de D3.js, pero no una anterior a ambas.
- Eres libre de usar cualquier función provista por D3.js, y usar cualquier elemento SVG y sus características (como `rect`, `circle`, `path`, etc...).

### 1.4. Ejemplo

A continuación se presenta un **ejemplo** de glifo ya implementado, para que tengas una idea del resultado final del ejercicio:

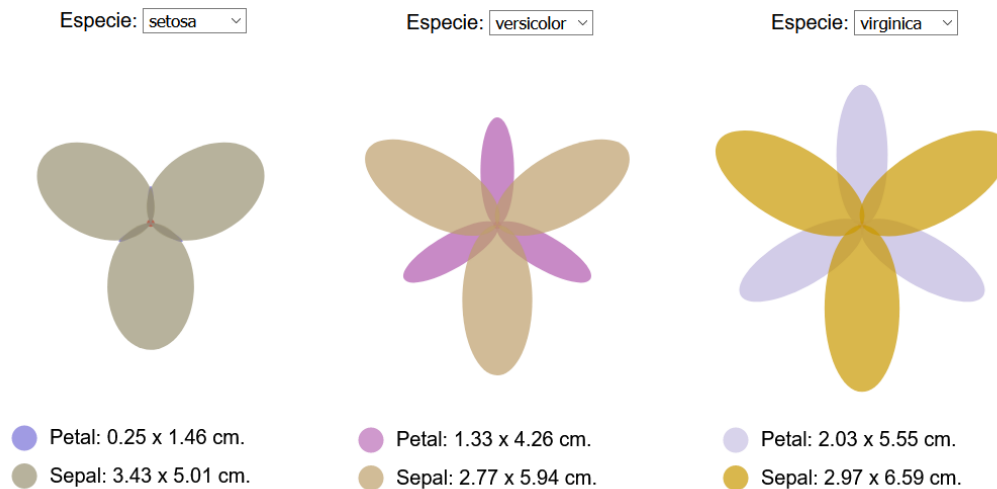


Figura 3: Ejemplo de glifo y su representación de cada especie de flor.

En el ejemplo anterior, se decidió representar cada pétalo y sépalo mediante [elipses](#), ubicadas de forma intercalada. Los radios de cada elipse son dictados por el valor del largo y ancho promedio de los pétalos y sépalos de cada especie de flor. La paleta de colores se hizo de forma independiente, y depende de la especie seleccionada. De esta forma, todos los atributos de cada elemento del *dataset* son representados en el glifo. Si lo deseas, puedes tratar de replicar este ejemplo en tu entrega o idear otro glifo distinto.

## 2. Descripción de proceso

Además de la generación de tu glifo, dentro de tu documento HTML, deberás incluir una **explicación de tu glifo** indicando de qué forma representas cada atributo de los datos entregados.

También deberás incluir un paso a paso de **cómo implementaste** la representación. En esta explicación deberás responder: ¿Qué utilidades de D3.js utilizaste? ¿Por qué esas?

Si no lograste realizar la visualización completa, puedes escribir sobre las partes que sí alcanzaste a implementar y cómo hubieras abordado lo que te faltó. También, comparte cuáles fueron tus **mayores dificultades** para esta entrega.

Para las tres secciones de descripciones pedidas, **no es necesario extenderse mucho**. Explicaciones concisas y efectivas son bien recibidas.

Dado que una de las metas de esta entrega es introducir a D3.js, es posible que tengan varias complicaciones o dudas respecto a lo que implementen o tengan que implementar. Es por esto que recomendamos poner cualquiera de esas dudas en el informe, para poder entregarles devuelta el mejor *feedback* posible y puedan aprovechar mejor su entrega. Esto también aplica al caso en que no puedan completar la totalidad de la evaluación: especifiquen que errores tienen o que cosas no lograron realizar como querían, y cuál creen que son las causas de las mismas.

### 3. Entregables

La entrega se realizará mediante la plataforma Canvas en la [evaluación correspondiente](#).

Para esta entrega se deberá subir un archivo de extensión **ZIP** que contenga un archivo de extensión HTML con todas las respuestas solicitadas anteriormente; y un programa de cargado de datos y generación de visualización JavaScript. Cualquier otro archivo (imágenes, hojas de estilo, *scripts*, etc.) que sea necesario para la visualización correcta de tu entrega debe también ir incluido en el archivo comprimido.

No se aceptarán entregas en cualquier otro formato distinto al indicado anteriormente (archivos PDF, TXT, DOC, etc...). Tampoco se aceptarán informes que sean imágenes de las respuestas insertadas en un archivo HTML.

### 4. Dudas

Cualquier duda que tengas sobre esta evaluación, prefiere publicarla en el [foro](#) correspondiente a esta evaluación. También siéntete libre de responder dudas de compañerxs si crees que manejas la respuesta.

### 5. Política de atraso

Las entregas tienen como fin preparar al estudiante para la instancia evaluativa que lo sigue (Hito). Es por esto que **no son evaluadas** y tienen un carácter completamente **opcional**. Sin embargo, la no entrega de alguna o entrega atrasada (por cualquier cantidad de tiempo) significará que el estudiante no recibirá *feedback* de su trabajo por parte del equipo docente. Este *feedback* tiene como finalidad corregir errores en la entrega para tener un mejor desempeño en el hito, por lo que se recomienda fuertemente la realización de este informe.