

Creación de una aplicación basada en grafos para LA VISUALIZACIÓN DE UN MODELO TAXONÓMICO (abril de 2020)

Jaime Acero, Jesús Ruiz y Andrés Villamizar, Estudiantes de Ingeniería en Multimedia

Resumen - Este documento es registro el avance realizado en un proyecto de aplicación de la computación gráfica en un programa que use la visualización de grafos como forma de mostrar información acerca de la clasificación taxonómica de una especie animal en concreto. Todo esto a través del uso de la librería d3.js y la teoría de la programación orientada a objetos.

Índice de Términos – Grafos, Taxonomía, Nodos

I. INTRODUCCIÓN

Aplicando la teoría de la computación gráfica en la creación de una aplicación que use los grafos como una forma de visualizar la información. Un grafo es el conjunto de objetos representados en nodos que están unidos a través de aristas que muestran relaciones entre estos. Se hace uso de los grafos ya que estos tienen la particularidad de permitir el estudio la relación que existe entre los distintos datos que lo conforman, creando así redes de información que se pueden comunicar entre sí.

Dado lo anterior se decidió aplicar la teoría de nodos a la creación de una red que almacena la información en base a la clasificación taxonómica de los animales. La Taxonomía es la ciencia que estudia los principios, métodos y fines de la clasificación, esta se aplica normalmente dentro de la biología para la ordenación jerarquizada de los grupos de animales y de vegetales.

II. OBJETIVOS

- Implementar los conceptos de la computación gráfica para hacer una aplicación que haga uso de los grafos como forma de mostrar la información al usuario.
- Investigar acerca del uso de la notación JSON para crear datos que se puedan comunicar entre ellos.
- Incluir el uso de los dispositivos de entrada para hacer que la aplicación tenga un componente interactivo en su uso

III. MARCO TEÓRICO

JavaScript: Es un lenguaje de programación enfocado principalmente la programación orientada a objetos con funciones de primera clase. Aunque es más conocido como el lenguaje de scripting para páginas web.

JSON: Es un formato basado en texto estándar para representar datos estructurados en la sintaxis de objetos de JavaScript. Es comúnmente utilizado para transmitir datos en aplicaciones web. [1]

Grafos: Es la composición de un conjunto de dos o más nodos que están conectados a otros nodos a través de conexiones conocidas como aristas; estos nos permiten conocer y relacionar las unidades que interactúan con otras, permitiendo analizar y solucionar diversas posibilidades o problemas a partir de sus datos.

[1]<https://developer.mozilla.org/es/docs/Learn/JavaScript/Objects/JSON>

[2]<https://www.ecured.cu/Taxonom%C3%ADa>

Taxonomía: Es una subdisciplina de la Biología Sistemática, que estudia las relaciones de parentesco entre los organismos y su historia evolutiva, actúa después de haberse resuelto el árbol filogenético de los organismos estudiados. [2]

D3.js: Es una biblioteca de JavaScript para producir, a partir de datos, infogramas dinámicos e interactivos en navegadores web. Hace uso de tecnologías bien sustentadas como SVG, HTML5, y CSS.

Nodos: Son los puntos principales de una red informática en los que se realiza una conmutación de líneas, estos tiene la característica de contener una serie de información la cual puede estar relacionada con uno o más nodos, el conjunto de estos elementos es conocido como grafo .

Url: Dirección asignada a cada uno de los recursos disponibles en la red que sean localizados e identificados. Creada por Tim Berners-Lee en 1991.

dat.GUI: Interfaz gráfica de controladores livianos para JavaScript. Dicha biblioteca fue desarrollada por Google Data Arts. la cual le permite modificar variables y observar los resultados sobre la marcha.

Algoritmo: Los algoritmos son múltiples instrucciones las cuales tienen el propósito de resolver un problema. Por lo tanto se toma como un conjunto de reglas que son aplicadas a una serie de datos, buscando resolver un problema en un número finito de pasos.

IV. REGISTRO DE AVANCES EN EL PROYECTO

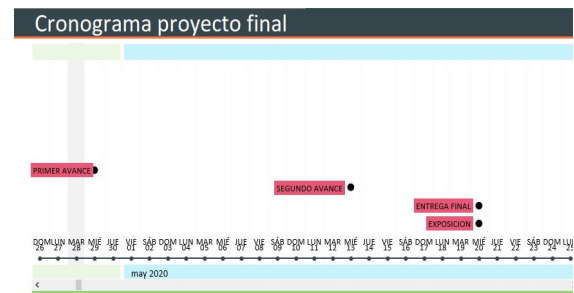


Fig. 1. Cronograma

Como se puede apreciar en la imagen, el proyecto se divide en 4 entregas, las cuales son:

1. Estado del arte

Se realizó el estado del arte del proyecto en el cual se explicó lo que buscábamos hacer y recolectamos información de proyectos similares basados en nodos y archivos .json, es esta parte se decidió que el programa tendría usaría tendría forma de árbol para representar la jerarquía que postula el modelo de clasificación taxonómico.

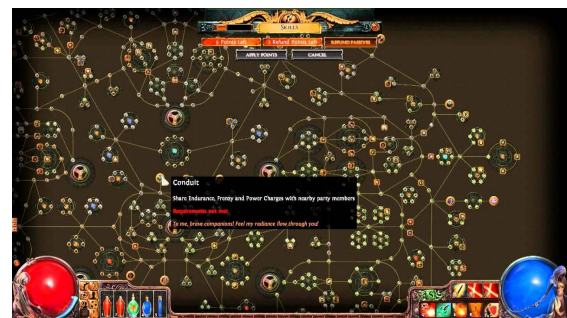


Fig. 2. Árbol de habilidades en Path of Exile.

2. Primera Avance

Realizamos la estructura base del programa basado en la la clasificación taxonómica de los animales usando nodos y notación JSON como forma de organizar la información dentro de la aplicación, esta es la estructura que buscaremos repetir para cada ecosistema que queremos abarcar en nuestra aplicación

```

var Mamiferos = [{
  "Orden": "Carnivora",
  "Familia": "Canidae",
  "Subfamilia": "Caninae",
  "Tribu": "Canini",
  "Genero": "Canis",
  "Especie": "Canis lupus familiaris",
  "Nombre Comun": "Perro",
  "Estado": "Domesticado"
},
{
  "Orden": "Carnivora",
  "Familia": "Canidae",
  "Subfamilia": "Caninae",
  "Tribu": "Canini",
  "Genero": "Canis",
  "Especie": "Canis lupus",
  "Nombre Comun": "Lobo Gris",
  "Estado": "Preocupacion Menor"
},
{
  "Orden": "Carnivora",
  "Familia": "Canidae",
  "Subfamilia": "Caninae",
  "Tribu": "Canini",
  "Genero": "Canis",
  "Especie": "Canis lupus",
  "Nombre Comun": "Lobo Gris",
  "Estado": "Preocupacion Menor"
}
]

```

Fig. 3. Ejemplo código JSON.

3. Segundo Avance

Desde el 29 de abril hasta el 13 de mayo se busca crear estructura representar la estructura en forma de árbol que tendrá el grafo para ello se empieza a investigar sobre librerías compatibles con JavaScript que se adapten a nuestra necesidades, en este punto decidimos usar la librería gráfica d3.js que permite crear todo tipo de estructuras que tiene como finalidad mostrar información en un entorno HTML

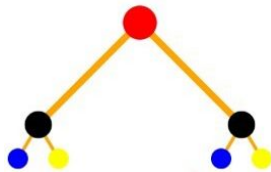


Fig. 4. Primera interpretación gráfica usando d3.js.

4. Tercer Avance

A partir del 13 de mayo hasta el 20 del mismo, se continúa trabajando la parte gráfica del programa con la librería d3.js, se busca agregar las interacciones con el mouse y con botones, por último se intentará pulir el lado estético del programa incluyendo animaciones, junto se empieza a incluir información a cada nodo entre lo que se incluye imágenes correspondientes del animal que este debe representar.

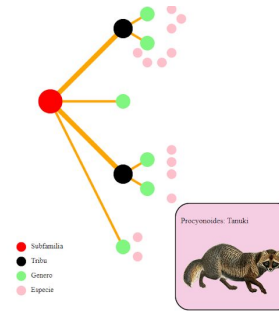


Fig. 5. Primer prototipo

5. Último prototipo

Teniendo ya la base de lo que queríamos hacer, pasamos a hacer el prototipo en 3D, lo primero fue hacer los arreglos de los nodos, dándoles su respectiva id, ubicación, categoría e información, partiendo de estos nodos creamos las esferas, las aristas y las unimos con los edges.

Teniendo ya el lado gráfico le dimos un color distinto a cada esfera, esto con el fin de diferenciar entre subfamilias, tribus, géneros o especies.

Para las interacciones del programa pensamos en que con el teclado el usuario pueda acercarse a cada esfera por lo cual, creamos una función que evalúa dónde está cada esfera y ubica la cámara en la esfera deseada.

Finalmente creamos pequeñas fichas bibliográficas las cuales se activan al darle click a la esfera de la que se quiera saber la información

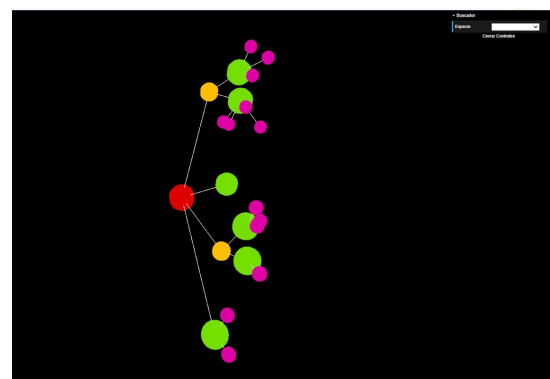


Fig. 5. Prototipo Final

la información que tiene el arreglo de nodos está pensado en que se puedan crear nuevos nodos y se pueda leer partiendo de una base de datos, sin embargo no implementamos esto porque no tuvimos el tiempo de investigarlo, sin embargo como todo el programa nace de este arreglo, solo faltaría guardar o cargar esta información.

Las texturas de las esferas, para hacerlas de forma eficiente se podrían hacer teniendo en cuenta las UV, sin embargo, no alcanzamos a hacerlas

VI. CONCLUSIONES

- Integramos la librería D3.js para poder representar de manera correcta el diagrama que conformará el grafo de nuestro programa.
- Incluimos el uso de dispositivos de entrada para darle un componente interactivo a nuestra aplicación
- Aplicamos el modelo taxonómico para organizar y dar información a los nodos que conforman el grafo.
- Analizamos el correcto uso de la notación JSON y como esta nos puede servir para optimizar el uso de la información.
- Implementamos algoritmos de búsqueda de caminos en grafos para visualizar información de manera rápida e interactiva.
- Usamos alojamiento de imágenes en línea para mostrar la información de cada nodo en el grafo, mejorando la funcionalidad de la aplicación ya que gracias a esta función el programa hace uso de múltiples ventanas.
- Mejoramos la interactividad del programa con el uso de la librería dat.gui.min.js, la cual permite el rápido acceso a funciones dentro de la aplicación.

A.. Referencias

- 1) NA. [d3Vienno].)2013, abril 9). D3.js tutorial [Archivo de video]. Recuperado de

https://www.youtube.com/watch?v=IJgEx_yb4u0

- 2) Mike Bostock. (2018). D3.js Graph Gallery. 2020, de the d3 graph gallery Sitio web: <https://www.d3-graph-gallery.com/index.html>
- 3) TheFreeDictionary (2013). TheFreeDictionary Sitio web: <https://es.thefreedictionary.com/nodos>
- 4) Graus (2013). Significados Sitio web: <https://www.significados.com/url/#:~:text=UR>
- 5) Google Open Source. Sitio web: <https://opensource.google/projects/datgui>
- 6) Sergio C. Fanjul. (2020). Retina. Sitio web: https://retina.elpais.com/retina/2018/03/22/tendencias/1521745909_941081.html