

Archivo de Documentación

para

Proyecto Corto 1

Preparado por:

**Paul Esteban Gutierrez Salas
2016183680**

**Carlos Andre Montero Moreno
2015067158**

Organización:

Instituto Tecnológico de Costa Rica

25 de Agosto del 2019

Índice

Índice	2
Introducción	2
Información Técnica del Proyecto	2
Bitácora	2
Referencias	3

Introducción

A continuación se muestra la documentación del Proyecto Corto 1 del Curso IC-6400 Visualización de Información.

Información Técnica del Proyecto

El proyecto consiste en tres partes:

- HTML: Este define qué recursos en general usamos, y dónde los colocamos, así como la forma del documento como tal.
- Javascript: Este define cómo se usaron los recursos que usamos, así como toda la mecánica interna de la página implementada.
- CSS: Este define cómo se visualizan las partes que usamos.

Para la sección de Javascript usamos la librería D3, con las partes de D3-tip, D3-figure y D3-svg.

En la sección de HTML colocamos una sección de tipo SVG vacía que se puebla y manipula utilizando métodos en el Javascript.

Para poblar el SVG se utilizó principalmente D3-Figure y la mecánica es agregar las figuras y modificar su forma según se cambia la visualización, esto se hace mediante el cambio en directo de atributos.

Bitácora

- 23 de Agosto: 2 Horas

Se crean los archivos iniciales para el proyecto y se crea un repositorio en Github para preservarlos. Este repositorio se encuentra en:

<https://github.com/UnopposedQuill/VIPC1>

- 24 de Agosto: 4 Horas

Se agregan los archivos de especificación dentro del repositorio. A nivel de proyecto se adquirió la librería D3, y se tomó la decisión de utilizarla mediante referencia a nivel web en lugar de descargarla con el objetivo de utilizar la más reciente. También se implementa el svg vacío para poder colocar la visualización creada con Javascript.

- 25 de Agosto: 9 Horas

Se tuvo problemas a la hora de dibujar los ejes del SVG de la visualización. Luego de unas horas se descubrió que fue debido a que previamente fue necesario desviar los ejes un poco para poder colocar el texto de cada uno de los ticks, una vez rastreado fue posible reajustar el tamaño de cada uno de los ejes para tomar un tamaño adecuado. Luego de terminar los ejes se implementó la visualización básica de las burbujas. Una vez se implementó, se invirtió un tiempo para averiguar cómo cambiar la forma de las figuras. Para esto se tuvo problemas con tamaños de figuras más pequeños, pues aparecen desviados a simple vista.

Se tomó la decisión de visualizar la información específica de cada uno de los cantones usando tooltips, que mostraran la información al colocar el mouse encima de la figura, se tuvo problemas a la hora de poder mostrarlos, al usar una librería disponible en:

<https://github.com/caged/d3-tip>

Esta librería no dio resultados debido a que estaba diseñada para ser usada con una versión anterior de d3. Se logró encontrar una modificación más nueva de d3-tip en:

<https://github.com/VACLab/d3-tip>

Más adelante conforme se fueron agregando nuevos valores de cantones, se descubrió que el label del tool-tip no acepta tildes.

Se tomó la decisión de colocar los datos de los cantones en un archivo de Javascript separado, así como de limpiar los comentarios.

Se corrigieron los labels sin tildes usando valores directamente de la tabla de Unicode encontrada en:

<https://www.utf8-chartable.de/unicode-utf8-table.pl>

- 26 de Agosto: 2 Horas

Se implementaron unos controles placeholders para verificar que el código funcionaba de manera correcta. Para esto se colocaron 3 botones que aprovechaban funciones en Javascript, las cuales, en función de muchas variables eran capaces de redibujar las pantallas.

- 28 de Agosto: 2 Horas

Se tuvo problemas a la hora de cambiar los controles, esto debido a que los controles nuevos dependían del CSS, el cual no se podía sobrescribir. Se sospecha que esto es debido a que no se atrapan los cambios en el form de los botones, se consideró la opción de usar Escuchadores de Eventos para solucionar esto, pero se tuvo que pensar en el rendimiento ya bajo de la página.

- 30 de Agosto: 2 Horas

Se le agregó una flecha al recuadro de despliegue de información de cada uno de los objetos desplegados y además se intentó agregar de manera no muy exitosa los glifos ya que estos no aparecen en la pantalla, también se intentó resolver el problema de mouse hover después del cambio de color de las figuras ya que esta se desactiva de manera misteriosa después del cambio de color pero con el cambio de tamaño su funcionamiento no se altera.

- 31 de Agosto: 10 Horas

Se tuvo problemas al realizar los cambios para incluir la figura de diamantes. Se tomó la decisión de empezar a usar el d3.svg.symbol para generar los símbolos de la visualización. Se utilizó el método descrito en:

<https://codepen.io/alsheuski/pen/PNWvel>

Se tuvo problemas con la visualización debido a un bug de d3 que elimina los dos primeros elementos de la lista de datos. Así que se decidió usar datos de sacrificio.

Se descubrió que el método de generador de symbol no tiene el símbolo "rectángulo", así que tuvo que tomar la decisión de si se hacían nuevos. Se decidió hacer un nuevo tipo de figura usando el método encontrado en:

<https://bl.ocks.org/iosonosempreio/0fd290e63eef4e2e709b318bdd2dfa52>

<https://bl.ocks.org/iosonosempreio/raw/0fd290e63eef4e2e709b318bdd2dfa52/script.js>

Se le empezó a configurar el css para que fuera más agradable a la vista del usuario usando fondos infinitos y haciendo los contenedores con un fondo transparente y eligiendo colores que no sean muy cansados para la vista. Los fondos infinitos se consiguieron en:

<https://www.toptal.com/designers/subtlepatterns/page/1/>

Se tuvo problemas al cambiar algunos aspectos del css cuando se inició a hacer los efectos. Estos incluyen problemas con el tool-tip, problemas de refresco, y otros relacionados con el escalado, entre los cuales está el problema crítico de que Tibas oculta completamente a Curridabat. Se agregó el efecto 3d usando:

<https://www.creativeblog.com/how-to/boost-d3js-charts-with-svg-gradients>

<https://stackoverflow.com/questions/24673627/d3-force-layout-how-to-achieve-3d-look-of-nodes>

Se agrego el efecto de brillos alterando la configuración de las figuras mediante la alteración de sus valores predeterminados.

Se cambió el css de la página con la finalidad de que esta fuera más atractiva visualmente para los usuarios y que a su vez no se viera monótona o muy simple.

Decisiones tomadas para el diseño visual

Lo primero que se puede decir es que se eligió un fondo infinito que a su vez ayuda que la apariencia de la interfaz no se vea tan monótona o simple.

Lo segundo que se puede apreciar es que se colocaron unos recuadros grises claros tanto al título como al cuerpos en donde desplegará la información. esto se hizo con el propósito de hacer una separación con el fondo de la página y que a su vez se pueda ver más ordenado y claro.

Tercero se puede apreciar que se colocó un footer con un fondo un poco más oscuro que los dos anteriores esto es para poder brindar la información de donde salió los datos que se utilizaron para realizar esta demostración gráfica.

Cuarto la forma que el usuario puede controlar la demostración gráfica es un recuadro gris que se encuentra en el lado superior de la demostración y que se aprovechó esta área ya que no se encontraba ningún punto por lo cual ninguno correría el riesgo de quedar oculto y a su vez se colocó en un lugar fácil de ver para el usuario.

Quinto la paleta de color utilizada en esta demostración gráfica fue principalmente una escala de grises que tiene como objetivo no ser muy cansada visualmente hablando ya que si hay muchos colores claros puede llegar a ser molesto para algunas personas y si se utiliza un colores muy oscuros este puede provocar una fatiga en los ojos al momento de utilizar una interfaz por mucho tiempo; por lo cual una escala de grises es más amigable para la vista del usuario.

Sexto el color inicial de los objetos es #80ff00 en escala hexadecimal o (128,255,0) en escala RGB ya que genera un color verde llamativo que a su vez da un contraste con el borde negro que las figuras tienen.

Referencias

<https://github.com/UnopposedQuill/VIPC1>

<http://bl.ocks.org/Caged/6476579>

<https://www.w3schools.com/js/default.asp>

<https://www.w3schools.com/css/default.asp>

<https://www.w3schools.com/html/default.asp>

<https://github.com/d3/d3/wiki/Tutorials>

<https://scrimba.com/g/gd3js>

<https://github.com/caged/d3-tip>

<https://www.utf8-chartable.de/unicode-utf8-table.pl>
<https://bl.ocks.org/d3indepth/bae221df69af953fb06351e1391e89a0>
<https://codepen.io/alsheuski/pen/PNWveL>
<https://github.com/d3/d3-shape/issues/64>
<https://bl.ocks.org/iosonosempreio/0fd290e63eef4e2e709b318bdd2dfa52>
<https://bl.ocks.org/iosonosempreio/raw/0fd290e63eef4e2e709b318bdd2dfa52/script.js>
<https://www.toptal.com/designers/subtlepatterns/page/1/>
<https://www.creativebloq.com/how-to/boost-d3js-charts-with-svg-gradients>
<https://stackoverflow.com/questions/24673627/d3-force-layout-how-to-achieve-3d-look-of-nodes>
<http://szudzik.com/ElegantPairing.pdf>
https://en.wikipedia.org/wiki/Pairing_function#Cantor_pairing_function