



Programación de Objetos Distribuidos

Informe del Trabajo Práctico Especial N°1

Integrantes del Grupo n°8:

- | | |
|----------------------------|-------|
| • Emilio Basualdo Cibils | 58172 |
| • Pedro Remigio Pingarilho | 58451 |
| • Miguel Di Luca | 58460 |
| • Fermín Gómez | 58111 |

Diseño e implementación de los servicios

Servicio de conteo:

STV nivel provincial: Implementamos una versión similar al modelo [Gregory](#)¹ del sistema de conteo STV de la siguiente manera;

La mínima cantidad de votos requeridos para ser un candidato ganador es igual a la cantidad de votos emitidos en la respectiva provincia, dividido por la cantidad de posibles ganadores².

A diferencia del STV original, cuando un candidato sobrepasa el mínimo, los votos sobrantes no son *transferidos*. En cambio, se agrupan los votos de este candidato en función de las elecciones de segundo o tercer orden³⁴, se cuenta la cantidad de votos para cada otro candidato, y se calcula el peso relativo⁵ de estos votos. Una vez considerados todos los candidatos que sobrepasan el mínimo, distribuimos los votos pesados e iniciamos el proceso de nuevo.

Si no hay ganadores, procedemos a eliminar aquel candidato que tiene la menor cantidad de votos(puede ser más de uno en casos de empate en la última posición) y, en este caso, transferimos los votos con su peso completo a los demás candidatos según corresponda. Y se repite el proceso.

De esta manera, el ganador tendrá siempre la misma cantidad de votos con las que pasó el mínimo, los votos de segundo o tercer orden tienen menos peso que los de primer orden, excepto en casos de eliminación, y es posible que la suma de porcentaje de votos entre todos los ganadores sea mayor a 100%, pues nunca se restan votos.

Si en algún momento hay menos o misma cantidad de candidatos que de sillas disponibles, estos candidatos serán considerados ganadores aunque su cantidad de votos no supere la mínima requerida.

En casos donde todos los perdedores se encuentran en la exacta misma situación y elegir un ganador es cuestión arbitraria/azar, optamos en elegir suficientes ganadores por orden alfabético.

AVT nivel nacional: Implementamos el modelo clásico del sistema de votos de segunda vuelta instantánea. Los votantes votan por preferencia de candidatos (máximo 3 candidatos). Al realizar el escrutinio se consideran inicialmente la primera preferenci. Si un candidato obtiene la mayoría absoluta de votos se lo proclama como ganador. Si ningún candidato obtiene la mayoría absoluta se procede a eliminar al candidato con menor cantidad de votos, sus votos son transferidos a la segunda preferencia de cada uno de sus votos. En caso de que haya más de un candidato con la menor cantidad de votos, los mismos también serán eliminados y sus votos transferidos. Este proceso se repite hasta que algun candidato obtenga la mayoría absoluta de votos.

¹ https://en.wikipedia.org/wiki/Counting_single_transferable_votes#Gregory

² mínima-cantidad-votos = votos/ sillas-disponibles.

³ Orden: Número de elección, en orden descendente por preferencia, dentro de la lista del voto emitido por un votante

⁴ Es decir, se considera la segunda elección y luego la tercera elección dentro de la lista de un voto. Se eligirá la primera de las dos que sea válida. Válida: el candidato aún no ganó, ni fue eliminado.

⁵ peso-relativo = (votos-candidato-ganador - mínima-cantidad-votos) /
votos-candidato-ganador

Trabajo concurrente

Para la manipulación de datos se utiliza una clase llamada Elections, esta contiene datos que pueden ser modificados por más de un thread al mismo tiempo. Debido a esto se tuvieron en cuenta los casos que podrían traer problemas, como al cambiar el estado de las elecciones, agregar votos a las listas, agregar nuevos fiscales, y se los protegió para evitar que múltiples threads modifiquen esta información al mismo tiempo, corrompiendo la información.

Para la lectura de votos parciales no se realizó ningún tipo de protección, ya que en caso de leer mientras se agregan votos la diferencia de lo que se lee en comparación con lo que se votó al momento es mínima o incluso nula. Para que la diferencia sea nula se pudo aplicar un mecanismo para que solo se pueda leer o escribir al mismo tiempo en las listas, pero esto podría ralentizar el proceso considerablemente sin obtener ningún gran beneficio.

Al momento de agregar los fiscales se utilizó un mapa en el cual se estableció como llave un par que contiene la mesa y el partido político y una lista de clientes como valor. El proceso se realiza de forma sincrónica, ya que al agregar dos clientes al mismo par se podrían dar dos casos, si la llave no existe se podría agregar un par con el primer cliente y a su vez se agrega al otro cliente con la misma llave, reemplazando el valor y eliminando al primer cliente de la lista. En caso de que la llave exista se estaría agregando clientes a la lista simultáneamente, pudiendo obtener resultados indeseados.

Puntos de mejora y/o expansión

En el caso de Fiscalización se consideró como una buena posibilidad llevar un recuento de los votos obtenidos y votos totales, de esta forma se podría visualizar cuántos votos se obtuvieron en esa mesa para el partido político deseado y el porcentaje de votos sobre el total de la mesa.

También se tuvo en cuenta como una mejora dejar de notificar a los fiscales a los cuales no se pudo establecer una comunicación una cierta cantidad de veces, almacenando al fiscal con la cantidad de veces que se produjo un error de conexión, este número volvería a cero al comunicarse correctamente, al llegar a una cierta cantidad establecida la idea era quitar al fiscal ya que se considera que este se desconecta.