

ANÁLISIS DORFMAN REGISTRADURÍA



DARON ANDRÉS MERCADO GARCÍA - A00395421 - INGENIERÍA DE SISTEMAS
ANDRÉS FELIPE CABEZAS GUERRERO - A00394772 - INGENIERÍA DE SISTEMAS
DAVIDE FLAMINI CAZARAN - A00381665 - INGENIERÍA DE SISTEMAS

DOCENTE ALEJANDRO MUÑOZ BRAVO

UNIVERSIDAD ICESI

FACULTAD DE INGENIERÍA

INGENIERÍA DE SOFTWARE 4

29 DE NOVIEMBRE DE 2024

1) Requerimientos no refinados..... 3

R2..... 3

R6..... 3

R8..... 3

R9..... 3

2) Particionamiento de primer nivel..... 4

3) Asignación de primer nivel..... 4

4) Requerimientos refinado (primer sub-especificación):..... 4

5) Segundo nivel de asignación de requerimientos a subsistemas..... 5

6) Sub-Especificación de requerimientos (Ordenado por Subsistemas)..... 5

Gestión de Cliente (GC):..... 5

7) Segundo particionamiento (final):..... 7

8) Asignación de tercer nivel:..... 8

1) Requerimientos no refinados

Análisis de requerimientos

R1. Los grupos de votantes por cada ciudad se agrupan por número de cédula en mesas de votación. Las mesas están ubicadas en colegios y universidades, y para la agrupación se tiene en cuenta la dirección de residencia que el votante tendrá que haber registrado previamente en el sistema central.

R2.

A cualquier ciudadano se le debe permitir consultar en qué sitio debe votar, por una interfaz de usuario sencilla, ingresando el documento de identidad.

R6.

Para efectos de simulación y prueba, cada cliente debe implementar un pool de N consultas para ejecutar concurrentemente, donde N lo podrá ingresar el usuario final al iniciar el programa cliente.

R8.

El servidor, en su rutina principal, debe ejecutar un ciclo en el que pregunte al usuario que digite un mensaje. El usuario digitalará una ruta que incluye el nombre de un archivo. Este archivo tendrá una lista de documentos de identidad (uno por línea) a los que se desea consultar la mesa de votación. Si la cantidad de líneas es menor a la cantidad de consultas que un computador puede realizar sin perder consultas ni abortarse, él mismo debe hacerlas. En caso contrario, debe distribuir secciones de la lista entre los clientes registrados en él, notificándoles como sus Observadores, para que cada cliente realice las consultas de los documentos de identidad que le correspondió, sobre el servidor, en el menor tiempo posible (el tiempo estimado deseable es máximo un segundo).

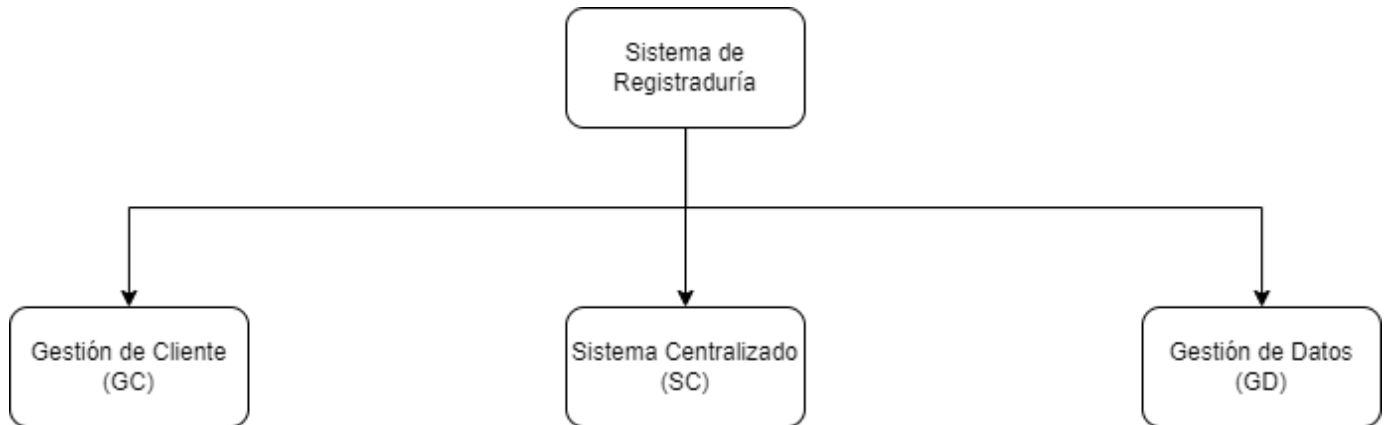
R9.

El servidor debe calcular el número de factores primos del documento de identidad y hacer la consulta en la BD de la mesa de votación. Si el número de factores primos es primo, debe retornar 1 si no, 0.

R10

El sistema debe ser auditable. En particular, en los clientes debe generarse un log con columnas separadas por coma. Las columnas son: la CC de consulta, y para cada CC, la respectiva mesa de votación, un indicador de si el número de factores primos es primo (1 si lo es, 0 si no lo es), y el tiempo que se demoró el sistema en responder, etc., que permita verificar la correctitud de las respuestas. En el servidor, debe dejarse la misma información por cada línea, pero antecedido del id del cliente que lo consultó, en un solo archivo de salida. En la última línea del archivo de salida, el número total de consultas realizadas y el tiempo total de ejecución.

2) Particionamiento de primer nivel



3) Asignación de primer nivel

	Gestión de Cliente (GC)	Sistema centralizado (SC)	Gestión de datos (GD)
R1.		X	X
R2.	X	X	
R6.	X		
R8	X	X	
R9.		X	X
R10	X	X	

4) Requerimientos refinado (primer sub-especificación):

R1.1: El sistema centralizado debe agrupar a los votantes por número de cédula en mesas de votación, teniendo en cuenta la dirección de residencia registrada previamente.

R1.2: El sistema debe almacenar y consultar las direcciones de residencia de los votantes, previamente registradas, para determinar su mesa de votación.

R2: A cualquier ciudadano se le debe permitir consultar en qué sitio debe votar, por una interfaz de usuario sencilla, ingresando el documento de identidad.

R6: El cliente debe permitir al usuario definir el tamaño de un pool de consultas concurrentes al iniciar el programa cliente.

R8.1: El sistema, dentro de un bucle, debe permitir al usuario digitar un mensaje, en el cual el usuario digitará la ruta que incluye el nombre de un archivo que tiene la lista de documentos de identidad

R8.2: El servidor debe recibir la ruta de un archivo que contiene una lista de documentos de identidad, deberá analizarla y si la cantidad de líneas supera el límite, el servidor debe dividir las

entre los clientes registrados, notificándolos para que realicen las consultas en el menor tiempo posible.

R9: El servidor debe calcular el número de factores primos del documento de identidad y hacer la consulta en la BD de la mesa de votación. Si el número de factores primos es primo, debe retornar 1 si no, 0.

R10.1: El cliente debe generar un archivo log separado por comas con los datos:

- documento de identidad
- mesa de votación
- si el número es factor primo (1 o 0)
- tiempo de respuesta

R10.2: El servidor debe registrar en su log la misma información con el id del cliente que realizó la consulta. además de incluir en la última línea el número total de consultas realizadas y el tiempo total de ejecución.

Nota: Conservamos la misma partición del primer nivel

5) Segundo nivel de asignación de requerimientos a subsistemas

	Gestión de Cliente (GC)	Sistema centralizado (SC)	Gestión de datos (GD)
R1.1		X	
R1.2			X
R2.	X	X	
R6.	X		
R8.1		X	
R8.2		X	
R9.		X	X
R10.1	X		
R10.2		X	

6) Sub-Especificación de requerimientos (Ordenado por Subsistemas).

Gestión de Cliente (GC):

GC_R2.1: El sistema debe permitir al ciudadano ingresar el documento de identidad mediante una interfaz de usuario sencilla.

GC_R2.3: El controlador debe validar el formato del documento de identidad ingresado por el usuario, una vez hecho, deberá enviarlo al servidor central.

GC_R6.1: El sistema Gestión de cliente debe permitir al usuario definir el tamaño de un pool de consultas

concurrentes al iniciar el programa.

GC_R6.2: El controlador debe configurar el tamaño del pool de consultas concurrentes y delegar la creación del mismo al sistema central.

GC_R10.1: El cliente debe generar un archivo log separado por comas con los datos:

- documento de identidad
- mesa de votación
- si el número es factor primo (1 o 0)
- tiempo de respuesta

Sistema Centralizado (SC):

SC_R1.1: El sistema centralizado debe agrupar a los votantes por número de cédula en mesas de votación, teniendo en cuenta la dirección de residencia registrada previamente.

SC_R2.2: El sistema debe procesar la consulta y retornar la información de la mesa de votación asociada al documento ingresado.

SC_R8.1: El sistema, dentro de un bucle, debe permitir al usuario digitar un mensaje, en el cual el usuario digitará la ruta que incluye el nombre de un archivo que tiene la lista de documentos de identidad.

SC_R8.2: La interfaz de usuario debe permitir al usuario final ingresar la ruta de un archivo que contiene una lista de documentos de identidad.

SC_R8.3: El controlador debe analizar el archivo recibido y, si la cantidad de líneas supera el límite permitido, dividir las entre los clientes registrados, notificándolos como observadores para realizar las consultas en el menor tiempo posible

SC_R9.1: El servidor debe calcular el número de factores primos del documento de identidad recibido.

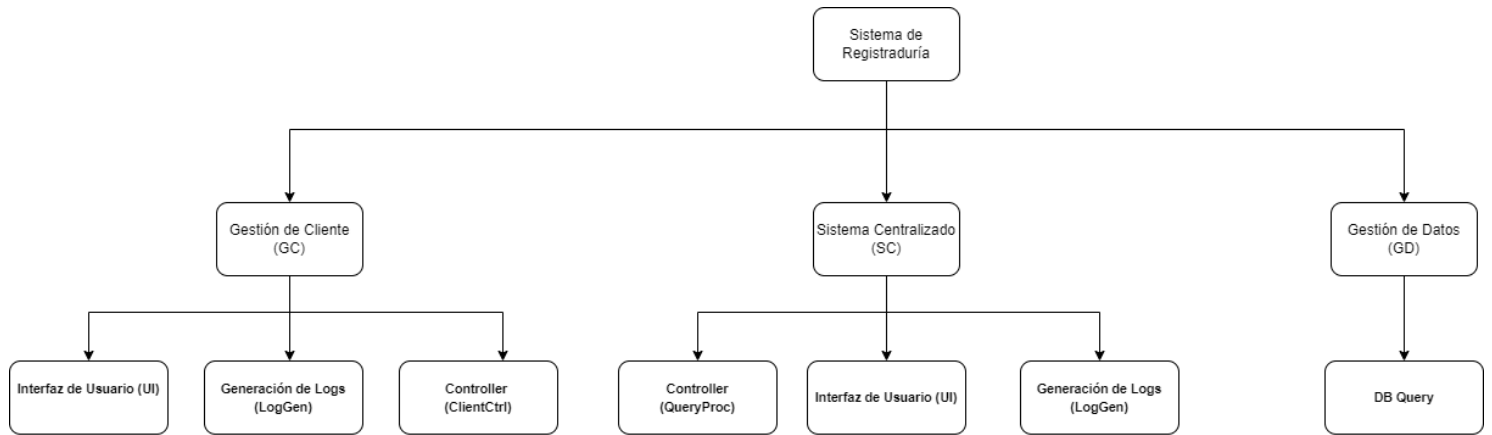
SC_R10.2: El servidor debe registrar en su log la misma información con el id del cliente que realizó la consulta. Además, debe incluir en la última línea el número total de consultas realizadas y el tiempo total de ejecución.

Gestión de Datos (GD):

GD_R9.2: El sistema debe consultar en la base de datos la mesa de votación correspondiente al documento de identidad. Si el número de factores primos es primo, debe retornar 1; si no, debe retornar 0.

GD_R1.2: El sistema debe almacenar y consultar las direcciones de residencia de los votantes, previamente registradas, para determinar su mesa de votación.

7) Segundo particionamiento (final):



8) Asignación de tercer nivel:

	Gestión de Cliente (GC)			Sistema centralizado (SC)			Gestión de datos (GD)
	Interfaz de Usuario (UI)	Generación de Logs (LogGen)	Controller (ClientCtrl)	Controller (QueryProc)	Interfaz de Usuario (UI)	Generación de Logs (LogGen)	DBQuery
SC_R1.1				X			
GD_R1.2							X
GC_R2.1	X						
SC_R2.2				X			
C_R2.3			X				
GC_R6.1	X						
GC_R6.2			X				
SC_R8.1					X		
SC_R8.2					X		
SC_R8.3				X			
SC_R9.1				X			
GD_R9.2							X
GC_R10. 1		X					
SC_R10. 2						X	