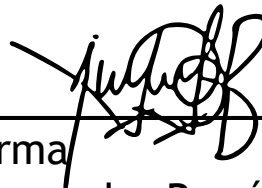


Proyecto FBCache



Firma
Luis Fuentes



Firma
Alexander Ramírez

7 de agosto de 2020



Pasantía Presencial: ☐

ó

Pasantía Virtual: ☒

Datos del Estudiante:

Apellidos, Nombres: Fuentes Salazar, Luis Reynaldo

Cédula de Identidad: V - 26.483.209

Teléfonos:

Celular: 0424-137-5135 Hab: 0212-561-6452

Correo electrónico: luisrf0540@gmail.com

Datos de la Empresa:

Nombre de la empresa: Synergy Vision

Dirección de la empresa: Centro San Ignacio, Torre Kepler, Ofic 603, Chacao.

RIF: J-40359909-2

Teléfonos: 0414-276-9752

Datos del Tutor Empresarial:

Apellidos, Nombres Tutor Empresarial: Ramírez, Alexander

Cédula de Identidad: 10.488.134

Teléfono Tutor Empresarial: 0414-276-9752

Correo Electrónico Tutor Empresarial: ar@synergy.vision

Tutor Empresarial es Prof. UCAB: ☐

Tutor Empresarial es (fue) tutor TEG ó TIG: ☐

Fecha de Inicio y Fin de la Pasantía: 24/08/2020 al 02/10/2020



Planteamiento del problema

Firestore es una plataforma en la nube propiedad de Google que ofrece distintos servicios para el desarrollo de aplicaciones, entre los servicios se encuentran 2 gestores de bases de datos, Firestore y Firebase Real Time Database; Google cobra a los clientes de estos por descargar la información que se encuentra almacenada en ellos, pero no cobra por la cantidad de información que se almacene, esto trae como consecuencia que el costo por mantenerlos dependa únicamente de la cantidad de consultas que se realicen. Si bien, que el costo dependa solamente de la cantidad de información que se extraiga se ve bastante atractivo, puede generar un costo muy variable en el mantenimiento; lo que puede traer como consecuencia que no se pueda hacer una estimación muy precisa de cuánto puede costar el servicio de almacenamiento, o que en un momento en específico se puedan generar muchas consultas que incrementen considerablemente el costo.

A todo esto, se suma que al ir creciendo la concurrencia de usuarios que usen alguna base de datos que se haya implementado en alguno de los gestores ofrecidos por Firestore, se van incrementando los efectos señalados anteriormente, llegando a ser un problema, ya que al no poder estimar bien el costo del mantenimiento del gestor de base de datos que se esté usando, y dicho costo poder ser muy elevado según el uso que se le dé, puede generar un pago excesivo.

La empresa Synergy Vision está en búsqueda de herramientas que permitan regular el costo del uso de Firestore y Real Time Database como gestores de base de datos, por lo que se quiere implementar una forma de disminuir la cantidad de consultas que se hacen directamente a estos, pero que no afecte a la disponibilidad que se tiene para acceder a la información almacenada.

Objetivo General

Desarrollar una librería en NodeJS que implemente un almacenamiento en caché de la información consultada los sistemas de gestión de datos Firebase

Objetivos específicos

1. Diseñar una librería pública para implementación de una caché de datos consultados sobre los sistemas de gestión de datos de Firebase
2. Diseñar un archivo de configuración donde se describa a qué rutas se le hará guardado en caché y especifique el tiempo de refrescamiento de la información almacenada en caché
3. Desarrollar métodos de consumo de la información almacenada en caché y refrescamiento de la información
4. Implementar un servidor de pruebas que muestre los logs de las consultas

Alcance

1. Diseñar una librería pública para implementación de una caché de datos consultados sobre los sistemas de gestión de datos de Firebase

Se diseñará una librería pública que se encargue de hacer la conexión a Firestore o Real Time Database, según sea el caso, realizar las consultas a las rutas que se le especifique, y almacenar en caché lo que se obtenga. Luego, cuando se realice otra consulta a una ruta especificada, la librería revisará lo que tenga almacenado, y resolverá esa solicitud. En caso de que se quiera consultar una ruta que no esté especificada, se hará una petición directa a base de datos para resolver la solicitud. La librería también tendrá la función de asignarle un tiempo de refrescamiento a la información que se almacene, el cual permitirá que se actualice cada cierto tiempo de forma automática.

Para este caso, el desarrollo de una librería pública implica que esta se va a alojar en un repositorio público, donde cualquiera que quiera probarla o usarla en un proyecto, pueda descargar el repositorio en donde se encuentre la librería e implementarla.



2. Diseñar un archivo de configuración donde se describa a qué rutas se le hará guardado en caché y especifique el tiempo de refrescamiento de la información almacenada en caché

Se diseñará un archivo de configuración para la librería, en el cual se le indique a esta, a qué rutas de Firestore o Real Time Database se les quiere hacer un almacenamiento en caché. Este archivo también indicará a la librería cada cuánto tiempo debe hacer una petición que permita actualizar lo que se tenga almacenado, cosa que permita tener la información actualizada con la menor cantidad de peticiones posibles; la configuración de la frecuencia de actualización de caché se puede hacer en segundos, horas, días, etc.

3. Desarrollar métodos de consumo de la información almacenada en caché y refrescamiento de la información

Se desarrollarán métodos para la librería que le permitan usar la información que tenga en caché para resolver consultas que originalmente se resolverían con una consulta directa a base de datos. También se desarrollarán métodos para que la librería pueda realizar automáticamente una petición directa a Firestore o Real Time Database para obtener los datos necesarios que permitan actualizar lo que esté almacenado; una vez se obtengan, se procederá a sobrescribir lo que se tenga almacenado con estos datos actualizados.

4. Implementar un servidor de pruebas que muestre los logs de las consultas

Se implementará un servidor donde se probará la librería. En este se desplegará la librería desarrollada para poder consultar y almacenar en caché la información de una base de datos implementada en uno de los gestores de datos de Firebase, especificada en el archivo de configuración; además, cada vez que se haga una consulta, el servidor debe mostrar logs por consola indicando si se resolvió usando la información almacenada en caché, o si se necesitó hacer una petición directa a base de datos.



Limitaciones

- Se utilizará NodeJS en su versión 12.18.3 LTS para desarrollar la librería, usando como lenguaje de programación JavaScript en su versión ECMAScript 2016
- El archivo de configuración de la librería tendrá un formato de tipo JSON
- El desarrollo se hará usando un control de versiones con Git, y será almacenado en un repositorio público en GitHub
- El servidor de pruebas será desplegado en Heroku
- La documentación del proyecto se encontrará en un archivo README.md en el repositorio de GitHub



Cronograma de trabajo

	S1	S2	S3	S4	S5	S6
Diseño de la librería						
Diseño del archivo de configuración						
Desarrollo de los métodos de consulta a Firebase						
Desarrollo de métodos de almacenamiento en caché						
Desarrollo de métodos de consulta a la información guardada en caché						
Desarrollo de métodos de refrescamiento de información almacenada en caché						
Implementación del servidor de prueba						
Despliegue del servidor en Heroku						

S1	24/08/2020 - 28/08/2020
S2	31/08/2020 - 04/09/2020
S3	07/09/2020 - 11/09/2020
S4	14/09/2020 - 18/09/2020
S5	21/09/2020 - 25/09/2020
S6	28/09/2020 - 02/10/2020

	Diseño
	Desarrollo
	Despliegue

Carta de Aceptación del Tutor Empresarial

Caracas, 7 de agosto de 2020


Señores
Consejo de Escuela de Ingeniería Informática
Facultad de Ingeniería
Universidad Católica Andrés Bello
Presente. -

Por medio de la presente hago constar que estoy dispuesto a supervisar, en calidad de Tutor Empresarial, la pasantía propuesta por: Luis Reynaldo Fuentes Salazar, C.I. 26.483.209, para lo cual solicito la aprobación de ese Consejo de Escuela.

Así mismo hago constar que estoy conforme con las responsabilidades que me corresponde asumir, las cuales se indican a continuación:

- Darle una inducción al pasante para adaptarlo a la organización.
- Dirigir y asesorar al pasante durante su permanencia en la empresa.
- Brindarle al pasante todas las facilidades relacionadas con: ubicación, materiales, obtención de información de otras unidades organizativas, etc.
- Evaluar el informe final de la pasantía, el estudiante debe entregármelo a más tardar una semana posterior a la fecha de finalización de la pasantía y el tutor se compromete a devolvérselo firmado junto con la Planilla de Evaluación Individual del Tutor Empresarial a más tardar una semana después.

Atentamente


Firma
Alexander Ramírez
Synergy Visión

