

Universidad Don Bosco
Diseño y Programación de Software Multiplataforma
DPS941



Estudiantes:

Ronald Gerardo Ramírez Guardado RG110604

Docente:

Ing. Alexander Alberto Sigüenza Campos

Contenido:

Foro 1: Bases de datos SQL y NoSQL

Fecha de entrega:
30 de abril de 2023

Estructura de Base de datos SQL

Base de datos SQL implementada en MySQL, específicamente en phpMyAdmin.

A continuación, se muestra la estructura de las diferentes tablas que comprenden la base de datos:

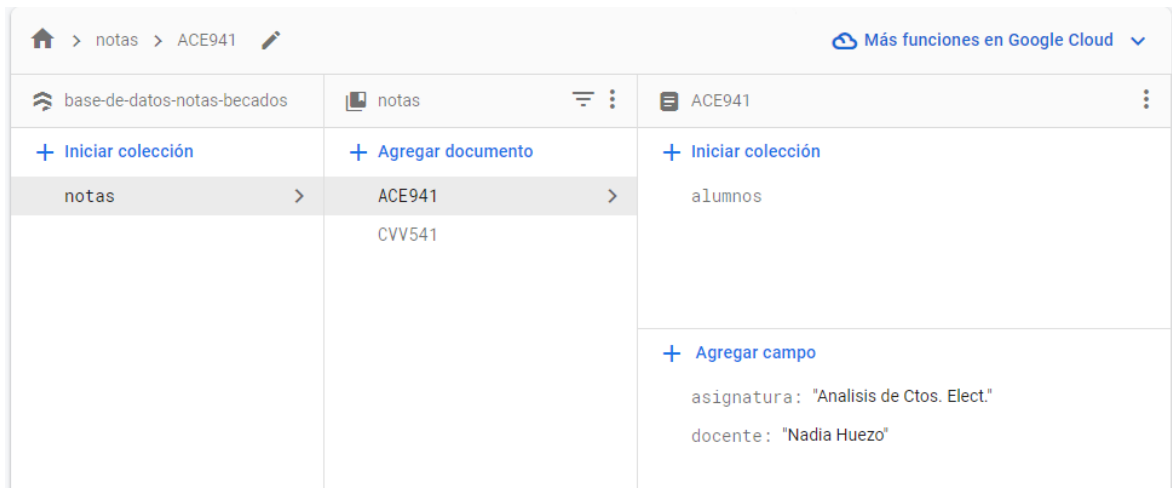
alumnos_becas carrera # id_carrera : int nombre_carrera : varchar(60)	alumnos_becas notas # id_alumno : int # id_asignatura : int # nota_final : float
alumnos_becas beca # id_beca : int nombre_beca : varchar(30)	alumnos_becas docente # id_docente : int nombre_prof : varchar(30) apellido_prof : varchar(30)
alumnos_becas alumno id_alumno : int nombre_alum : varchar(30) apellido_alum : varchar(30) carnet : varchar(10) # id_beca : int # id_carrera : int	alumnos_becas asignatura # id_asignatura : int cod_asignatura : varchar(8) nombre_asig : varchar(30) # id_docente : int

Podemos observar que se relacionan los diferentes ID de las distintas tablas para poder completar la información que el usuario requiera en las consultas que realice respecto a las notas del alumno becado.

Estructura de base de datos NoSQL: Firestore.

Para la base de datos NoSQL, se ha optado por Firestore, perteneciente a Firebase.

Específicamente al ser una base de datos no relacional y orientada a documentos JSON, se ha optado por una estructura de datos basada en subcolecciones donde, asumiendo que es ya sabido que los datos que deseamos obtener pertenecen a estudiantes becarios de UDB Virtual, se ha especificado “notas” como colección raíz.



Dicha colección contiene los documentos que harán referencia a la asignatura de la cuál se desean obtener las notas, a dicho documento pertenecerán los campos de “asignatura” el cuál especificara el nombre de ésta, y mostrarán el nombre del docente a cargo mediante el campo “docente”, de igual manera su ID será el código único de materia que es definido por la universidad.

Para cada documento que contenga la información de las asignaturas tendremos una subcolección llamada alumnos. Dicha subcolección mostrará la información de los diferentes alumnos inscritos en la asignatura impartida, de igual manera mostrará la información básica estudiantil, así como la beca a la que el alumno pertenece, todo esto será alojado en el documento identificado con el ID que la universidad asigna a cada alumno a través de su número de carnet, tal como se muestra a continuación:

🏠 > notas > CVV541 ☁ Más funciones en Google Cloud		
📁 base-de-datos-notas-becados	📁 notas	📁 CVV541
+ Iniciar colección	+ Agregar documento	+ Iniciar colección
notas >	ACE941	alumnos
	CVV541 >	
		+ Agregar campo
		asignatura: "Calculo de Varias Variables"
		docente: "Joel Orellana"

🏠 > notas > CVV541 > alumnos ☁ Más funciones en Google Cloud		
📁 notas	📁 CVV541	📁 alumnos
+ Agregar documento	+ Iniciar colección	+ Agregar documento
ACE941	alumnos >	GC152222
CVV541 >		MP223085
	+ Agregar campo	
	asignatura: "Calculo de Varias Variables"	
	docente: "Joel Orellana"	

🏠 > notas > CVV541 > alumnos > MP223085 ☁ Más funciones en Google Cloud		
<div>CVV541</div> <div>+ Iniciar colección</div> <div>alumnos ></div> <div>+ Agregar campo</div> <div> asignatura: "Calculo de Varias Variables" docente: "Joel Orellana" </div>	<div>alumnos</div> <div>+ Agregar documento</div> <div>GC152222</div> <div>MP223085 ></div>	<div>MP223085</div> <div>+ Iniciar colección</div> <div>+ Agregar campo</div> <div> apellido: "Portillo" carnet: "MP223085" carrera: "Ing. Industrial" nombre: "Maziel" nota: "6.3" </div>

Como se puede observar, en el mismo documento será almacenada la nota obtenida por el alumno en la asignatura correspondiente.

Conclusiones

La adaptación a las bases de datos no relacionales (NoSQL) puede llegar a ser difícil cuando se nos viene estandarizando el uso de las bases de datos SQL, sin embargo, si se saben aprovechar las ventajas de NoSQL se puede encontrar una manera mucho más fácil de administrar los datos alojados en éstos.

A manera de conveniencia en lo que a cambios o actualizaciones de datos se refiere, SQL es una opción muy sencilla, pues al relacionar las tablas que comprendan la base de datos, esta acción es tan sencilla como solo modificar el campo con llave primaria o foránea para que el cambio se realice también en las tablas que están relacionadas a estas llaves.

Sin embargo, como se mencionó al principio, si se saben aprovechar las ventajas de NoSQL, puede ni siquiera notarse la falta de esta característica, pues al implementar diferentes comandos para las estructuras no relacionales se puede obtener el mismo resultado de actualización de datos que en una base de datos SQL. De igual manera solo el hecho de que en base de datos como Firebase se tenga un almacenamiento en la nube, facilita muchas funciones que van también de la mano con la necesidad de acceder y modificar datos en tiempo real, y es aquí donde se puede afirmar el porqué es más conveniente el uso de Firebase en lugar Realtime para su implementación en React Native y es eso: LA GESTIÓN DE DATOS REALIZADAS EN TIEMPO REAL.