Cloud Computing (CS495)

Firebase Realtime Database

Lab #3

Prepared by:

Ashraf Mohey

➢ What is “Firebase Cloud messaging”?

Store and sync data with our NoSQL cloud database. Data is synced across all clients in realtime, and remains available when your app goes offline.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

➢ How does it work?

The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code. Data is persisted locally, and even while offline, realtime events continue to fire, giving the end user a responsive experience. When the device regains connection, the Realtime Database synchronizes the local data changes with the remote updates that occurred while the client was offline, merging any conflicts automatically

The Realtime Database provides a flexible, expression-based rules language, called Firebase Realtime Database Security Rules, to define how your data should be structured and when data can be read from or written to. When integrated with Firebase Authentication, developers can define who has access to what data, and how they can access it.

➢ Firebase Database vs MySQL vs SQLite?

Name	**Firebase Realtime Database** X	**MySQL** X	**SQLite** X
Description	Cloud-hosted realtime document store. iOS, Android, and JavaScript clients share one Realtime Database instance and automatically receive updates with the newest data.	Widely used open source RDBMS	Widely used in-process RDBMS
Primary database model	Document store	Relational DBMS	Relational DBMS
Secondary database models		Document store Key-value store	Key-value store
DB-Engines Ranking Trend Chart	Score 6.48 Rank #44 Overall #7 Document stores	Score 1223.34 Rank #2 Overall #2 Relational DBMS	Score 115.45 Rank #11 Overall #7 Relational DBMS
Website	firebase.google.com/products/-realtime-database	www.mysql.com	www.sqlite.org
Technical documentation	firebase.google.com/docs/-database	dev.mysql.com/doc	www.sqlite.org/docs.html
Developer	Google	Oracle	Dwayne Richard Hipp
Initial release	April, 2012	1995	2000
Current release		8.0.11, April 2018	3.23.1, April 2018
License	commercial	Open Source	Open Source

➢ Firebase Database vs MySQL vs SQLite?

Cloud-based	yes	no	no
DBaaS offerings		Google Cloud SQL: A fully-managed database service for the Google Cloud Platform	
Implementation language		C and C++	C
Server operating systems	hosted	FreeBSD Linux OS X Solaris Windows	server-less
Data scheme	schema-free	yes	yes
Typing	yes	yes	yes
XML support	no	yes	no
Secondary indexes	yes	yes	yes
SQL	no	yes	yes
APIs and other access methods	Android iOS JavaScript API RESTful HTTP API	ADO.NET JDBC ODBC	ADO.NET JDBC ODBC
Supported programming languages	Java JavaScript Objective-C	Ada C C# C++ D Delphi	Actionscript Ada Basic C C# C++

➢ Firebase Database vs MySQL vs SQLite?

For full comparison:
https://db-engines.com/en/system/Firebase+Realtime+Database%3BMySQL%3BSQLite

➢ Firebase Database Pro’s vs Con’s

Pro’s:

1. If your app does run of a centralized DB, and is updated by a lot of users – then it’s more than capable of handling the Real-Time data updates between devices.
2. Stored in the cloud so readily available everywhere.
3. Cross Platform API (If you are using this DB with an App)
4. They Host the data. -Meaning if you are storing a lot of data, you don’t have to worry about hardware!
5. Offline: Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is reestablished, the client device receives any changes it missed, synchronizing it with the current server state.
6. Realtime: Instead of typical HTTP requests, the Firebase Realtime Database uses data synchronization—every time data changes, any connected device receives that update within milliseconds. Provide collaborative and immersive experiences without thinking about networking code.

➢ Firebase Database Pro’s vs Con’s

Con’s:

1. Unless your app runs of one centralized database updated by a vast quantity of users, it’s a major overkill.
2. Storage format is entirely different to that of SQL, (Firebase uses JSON) so you wouldn’t be able to migrate that easily.
3. Reporting tools won’t be anywhere near the ones of standard SQL.
4. Costs! -Limited to 50 Connections and 100mb of Storage!
5. You don’t host the data, Firebase does. And depending on which server you get put on, viewing there up time there seems to be a lot of disruption lately.

➢ Configuring Android Project with Firebase Database

• Use the Firebase Assistant

To open the Firebase Assistant in Android Studio:

- ➢ Click **Tools > Firebase** to open the **Assistant** window.
- ➢ Click to expand firebase realtime database
- ➢ Click the **Connect to Firebase** button to connect to Firebase and add the necessary code to your app.

➢ Read/Write data to database

To read/write data to firebase in any client you always need to follow these steps:

- ➢ Get reference to your root object of your database.
  - ➢mDatabase = FirebaseDatabase.getInstance().getReference();
- ➢ Navigate inside your nodes using .child(‘’)
  - ➢mDatabase.child(‘NestedObject1’).child(‘NestedObject2’); ….
- ➢ Use nodeReference.setValue(Object) for write operation
- ➢Attach a listener for you node reference for read operations

➢ Read/Write data to database

For full usage reference please follow official documentation:
https://firebase.google.com/docs/database/android/read-and-write

➢ Task 3

➢ You are required to integrate Firebase database into your previously created android app in task 2 by following this tutorial
https://firebase.google.com/docs/database/android/start/

➢ When sending push notification from FCM Console, you are required to extract the content message and store in the database under "message" object like this
https://justpaste.it/1j3tx

❖Note: that the datetime format is "yyyy-MM-dd hh:mm:ss" or any other...

➢To receive the notification message and extract data from it check this tutorial
https://firebase.google.com/docs/cloud-messaging/android/receive

❖Note: Your app must be in foreground to receive the notification at onMessageReceived. you can use Logs to monitor this behavior

➢ Team is max 3 members

➢ Reference

https://firebase.google.com/docs/database/