Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 157 (2019) 396–405

Procedia Computer Science

www.elsevier.com/locate/procedia

4th International Conference on Computer Science and Computational Intelligence 2019 (ICCSCI), 12–13 September 2019

The Comparison Firebase Realtime Database and MySQL Database Performance using Wilcoxon Signed-Rank Test

Margaretha Ohyver[a]*, Jurike V. Moniaga[b], Iwa Sungkawa[a], Bonifasius Edwin Subagyo[b], Ian Argus Chandra[b]

[a]*Statistics Department, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia* 11480
[b]*Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia* 11480

Abstract

This study aims to compare Firebase Realtime Database and MySQL performance as Database Management System for our Toddlers Daily Nutritional Needs Mobile Application. The comparison is in terms of Database response time. Their performance is compared using a common CRUD operations. Each operations data is analyzed using Wilcoxon Signed-Rank test. Our comparison results confirm that Firebase Realtime Database is more suitable as our Daily Nutritional Needs Mobile Application *Database Management System* since it could perform better response time. The key features match our needs to provide Realtime Data and it is also help simplify the process of developing applications.

Keywords: Nutritional Status; Firebase Realtime Database, Mobile Application, MySQL Database, Wilcoxon signed-Rank Test;

1. Introduction

The database is a part of our daily life in the present which we are not often mindful. There are many data in this world that related each other so we need a system to do the data processing. Those data are called database. We need

* Corresponding author. *E-mail address:* mohyver@binus.edu

1877-0509
10.1016/j.procs.2019.08.231

a system to manage it and the system to do this is called Database Management System (DBMS) [1]. It simplifies users to store, obtain, and modify data in the database. When accessing data, user wants that the application program can provide fast response time. With fast response times user can use application more effectively, otherwise with slow response time user waste more time and the focus on the work become distracted. There are several DBMS that we can use. To decide which one is better, we can compare the response time of these databases. A database which has fast response time is the effective database. So, what is the real definition of database response time? It is the length of time required by the database to send a request until receiving the response from the server [2].

MySQL is a relational database server that supports the well-known SQL (Structured Query Language) database language [3]. According to Dataconomy, the latest version of MySQL is one of the world's popular databases. It is because MySQL is an open source database which reliable and compatible with all major hosting providers. It is also a cost-effective database which also easy to manage [4]. MySQL also has several shortcomings in term of scaling, for examples long development time and database logging costs. To help with additional scale problems, there are other alternatives databases such Firebase Realtime Database. It is a cloud-hosted NoSQL-based database. The Realtime provides syncing across connected devices and is available when there is no network connectivity through a local cache [5]. The Firebase Realtime Database is built under Google infrastructure, so developer without backend expertise can build apps fast without worrying too much about scalability and just focus on building the best apps [6]. The drawback of this database exists when the developer needs to query complex data and to migrate data. With the advantages and disadvantages of these two databases, we need to do further comparison to determine which database is more suitable for the development of our application.

This research was conducted by comparing Firebase Realtime Database and MySQL through our Mobile Application for Toddlers Daily Nutritional Needs. The details about this mobile application will present in next section. There are some researches have conducted in term of comparing databases. For examples, Truică et al (2015), Tang and Fan (2016), and Pereira et al (2017). Truică et al examined the execution time for CRUD (Create, Read, Update, and Delete) operations of MongoDB, CouchDB, and Couchbase. Here are the results. First, MongoDB is the fastest in fetching data. Second, CouchDB is a good choice for doing *intensive write operations*. Third, in term of data processing, Couchbase is a good choice [7]. Tang and Fan evaluated the performance of Redis, MongoDB, Couchbase, Cassandra, and HBase. They used YCSB as their measurement tool. The Yahoo! Cloud Serving Benchmark (YCSB) is an open-source NoSQL benchmarking system and it is often used to compare relative performance of NoSQL Database Management Systems. Performance, elasticity, availability and replication, are the aspects of evaluation. Redis has the best performance in insert and execute operations compares to the other databases. Although it shows the best efficiency, Redis also has its limitation when in the face of extremely large data [8]. Pereira et al evaluated features comparison and performance of Couchbase, MongoDB and RethinkDB. Tests were performed in two different scenarios: single thread (evaluate the database responsiveness) and multiple threads (evaluates the database performance in a load test scenario). The results reveal that Couchbase had a better performance at most of the operations, except for retrieving multiple documents and inserting documents with multiple threads, operations in which MongoDB scored better [9].

2. Mobile Application for Toddlers Daily Nutritional Needs

Malnutrition for toddlers is one of the main problems that occur in the world, including in Indonesia [10]. Therefore, the Government and the community must overcome this issue. One thing that we, as the part of community, can do is help the parents to monitor their children's nutritional status and provide nutritional knowledge by create mobile application. The application has created and named Toddlers Daily Nutritional Needs. This application is the follow-up of the future work of Ohyver et al (2017).

The Toddlers Daily Nutritional Needs help parents to check nutritional content of each food ingredient to be used such as calories, carbohydrate, protein, and fat. For parents that confused what to cook for their children, we also provide the list of food recommendation. The list of food had been adopted from the book which written by food nutritionists, Ayu Bulan Febry, K.D., S.KM. To be able to store all the data needed in the application, it is necessary to have a *Database Management System*.

3. The Wilcoxon Signed-Rank test

For parametric tests there are a few assumptions must fulfill. If one of the assumptions are not meet, then nonparametric tests are the solution. one of those tests is the Wilcoxon Signed-Rank test.

Wilcoxon Signed-Rank test is useful in comparing paired observations which come from two populations. The null hypothesis is that the median difference between the two populations is zero. The alternative hypothesis is that is not zero. The alternative hypothesis may also be the median of one population is less (or greater) than the median of the other population. The first step to do this test is we make a list which comprises the pairs of observations that we have on the two variables. For each pair, we compute the difference,

$$\boldsymbol{D = x_1 - x_2}. \tag{1}$$

Take the absolute values of *D* and rank it. Compute $\boldsymbol{\Sigma}(+)$ as the sum of the ranks of the positive differences. Also compute $\boldsymbol{\Sigma}(-)$ as the sum of the ranks of the negative differences. Here is the test statistic of Wilcoxon Signed-Rank test.

$$\boldsymbol{T = min}[\boldsymbol{\Sigma}(+), \boldsymbol{\Sigma}(-)] \tag{2}$$

The decision rule of this test is equal with the decision rule of Z test or t test [11].

4. Firebase Realtime Database Features

4.1. NoSQL

Data in Firebase Realtime Database does not consist of tables, but only in the form of JSON documents. Data in the NoSQL database can be placed on a different server. The cost of adding a server is much cheaper than upgrading the capacity of a server. NoSQL type of database supports Auto Sharding **[12]**. Data will automatically be balanced from various servers to the server pool, if there is one server down, the data can be directly migrated to another server. The possibility for server down is smaller compared to SQL.

4.2. Realtime Database

According to its name, Firebase Realtime Database is realtime this means that if one user who uses the application updates data, then the data on the Google Server will be updated immediately, and the Firebase system will immediately update all data on all other user that are using the application **[13]**.

4.3. Simplifies Backend Development

Firebase Realtime Database only requires that we make a code to change the database on the client side (web, Android application, iOS application, etc.). While *SQL* usually requires us to create server-side code with server programming languages such as PHP, Ruby, etc.

5. MySQL Database Features

5.1. Relational Database

A database consists of many tables. This table consists of many fields which are columns. Fill in each row of this table as data. To create an integrated database system, between one table and another table has a relationship that must always be maintained. Each table has a primary key, this primary key is then linked to the second table and becomes

a foreign key for this second table. With this relational database, the data will be consistently stored in a table, then other tables that need other data can only be connected via the foreign key.

5.2. Non-Realtime Database

SQL requires some configuration before you can access data in real time, while on Firebase Realtime Database, the system synchronizes data automatically.

5.3. Query Friendly

SQL databases can handle complex queries. Developer does not have to design a scheme for particular purpose. SQL databases have an enormous flexibility in query language, so it can return any data in any format without modify the underlying scheme. *NoSQL* databases are designed specifically to handle simple low level queries that allow you to get, set and a few other simple things like ranges or sorting. A big disadvantage on *NoSQL* is that Developer have to decide how the data you put in there and design a suitable schema up front to access the data.

6. Method

To compare the performance of MySQL Database and Firebase Realtime Database, the test phase done by performing CRUD operations on both MySQL Database and Firebase Realtime Database on the Toddlers Nutritional Status Application. For each of the databases, the CRUD operations performed in 50 times. CRUD operations are basic functions for retrieving and returning data from a database. Table 1 illustrates the software versions installed for testing.

Table 1. Database Versions.

Database	Version
Firebase Database	16.0.3
MySQL Database	5.7.25

Tests performed using experimental setup based on server and client configurations. The device used in the client test listed in Table 2. The study targets the latest version of Android in order to provide an up-to-date comparison of both databases and the operating system. The device the experiments evaluated on uses the latest version of Android, Android 9.0. In addition, for the machine used in the server side listed in Table 3.

Table 2. Hardware & Software Specification of the Device Used

Android Version	API Level	CPU	RAM	Network Speed
Android 9.0 (Pie)	28	Snapdragon 845 2.8GHz	6 GB	Full

The Data Definition Language (DDL), which deals with database schemas and descriptions, of how the data should reside in the database that used in this test was CREATE, to create a database and its objects like (table, index, views, store procedure, function, and triggers). For the SQL used in this test are the MySQL Database which is a database computer language designed for managing data in relational database management systems (RDBMS).

Table 3. Hardware Specification Of The Server Machine Used

CPU	RAM	Network Speed
Intel Xeon Quad Core	4 GB	Full

The table structures for SQL and JSON are using one table, which consists of a six toddler's nutrition status data: Toddler's Weight, Toddler's age, Toddler's nutritional score, Toddler nutritional status, Toddler unique id and Timestamp when the database saved. Figure 1 presents a datasets example that will used. The databases comparison

involved testing performance time for all the CRUD operations, each batch of tests run 50 times from one to 3,000 records. The experiments tested the performance of:

A. *CREATE Operation*
The create operation measures the time taken to add each data. To create a new record in MySQL Database by using *INSERT* method and in Firebase using *setValue()* method.

B. *READ Operation*
The read operations measure the time taken to select and return all the data, including the data added by the create operation in both database system. For Firebase this is done by using *ValueEventListener* method and. For MySQL Database it's done by using SELECT method.

C. *UPDATE Operation*
The update operation measures the time taken to correct each data. For Firebase this is done by directly updating the field using *setValue()* method. For MySQL Database it required a *SELECT* operation that gets all the information and then *UPDATE* can be performed.

D. *DELETE Operation*
The delete operation measures the time taken to delete each data. This operation deletes all data in the database by using *removeValue()* method in Firebase and *DELETE* method in MySQL Database.

```
{
  "child_id": "LXY8GZSoX84ExkwNaEy",
  "child_weight": 5,
  "child_age": 9,
  "status_score": 1.2,
  "status_gizi": "Gizi Baik",
  "date_save": "1-1-2019"
}
```

Fig. 1. Toddler's nutritional status sample database.

7. Results & Discussion

The multivariate Shapiro-Wilk test was conducted to determine the normality of data. Because the $p - value = 3.189 \times 10^{-8} < alpha$, we rejected the null hypothesis. It means that the data are not met the normality assumption. Based on Section 3, we use the Wilcoxon Signed-Rank test to analyze data. This test was conducted to each of CRUD operations.

7.1. CREATE Operation

In accordance with the CRUD operation process described in the method section, we loaded records into database during *CREATE* operation. Range of the records are from 1 to 3000 data. Figure 2 presents the performance results between Firebase Realtime Database and MySQL Database of *CREATE* operation. Y Axis shows response time in millisecond and X Axis shows the experiments. From the graphic illustrated that MySQL produce slower response time than Firebase Realtime Database. For example, we can see in experiment 5, there are 20 records. The response time for MySQL is 3218 milliseconds and for Firebase is 37 milliseconds. Until last experiment the response time of MySQL is always slower than Firebase Realtime Database.

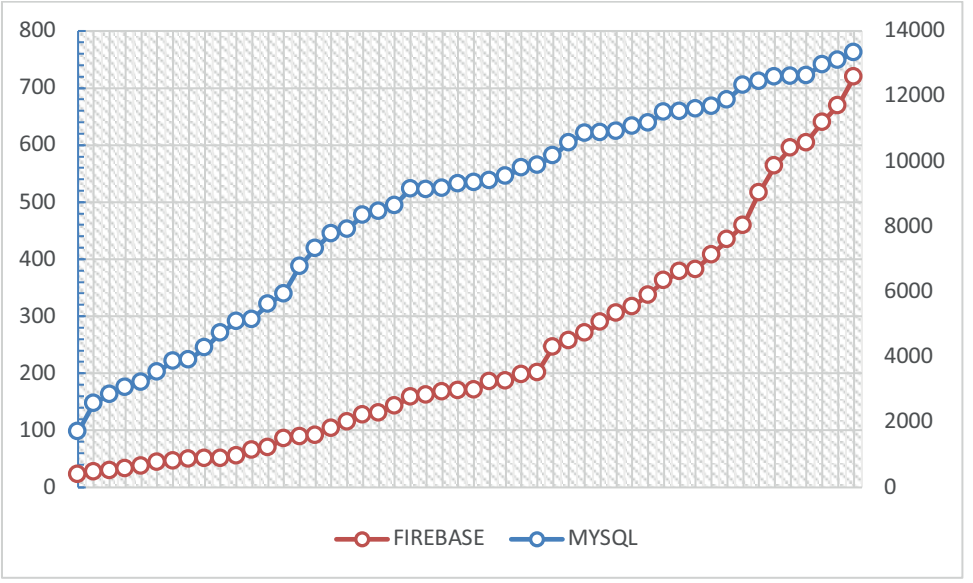

Fig. 2. Mean performance time in milliseconds for the *CREATE* operation

Here is the hypothesis for CREATE operation.

$$H_0: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ CREATE\ operation\ is\ worse\ than\ or\ equal\ to\ MySQL\ Database$$

$$H_1: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ CREATE\ operation\ is\ better\ than\ MySQL\ Database$$

The p-value for this test is 3.895×10^{-10}. Because the p-value < alpha, we reject the null hypothesis. It means that the performance of Firebase Realtime Database is better than MySQL Database in term of CREATE operation. This process also represents application responsiveness when parents would like to store their children data, measurement results of their children nutritional status, parents data, and daily toddlers food list on our Toddlers Daily Nutritional Application.

7.2. READ Operation

Figure 3 presents the performance results between Firebase Realtime Database and MySQL Database of *READ* operation. Y Axis shows response time in millisecond and X Axis shows the experiments. From the graphic illustrated that MySQL produce slower response time than Firebase Realtime Database.

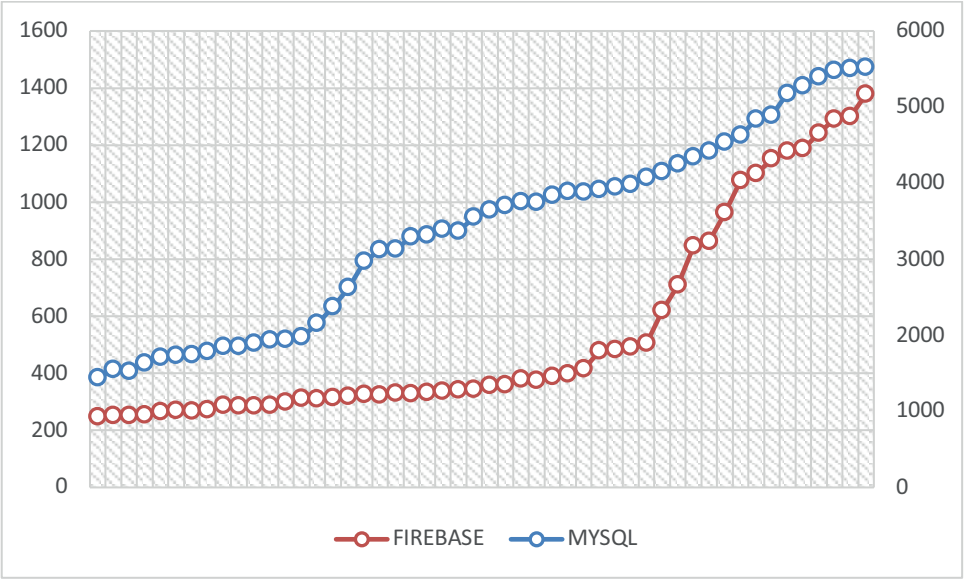

Fig. 3. Mean performance time in milliseconds for the *READ* operation

Here is the hypothesis for READ operation.

$$H_0: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ READ\ operation\ is\ worse\ than\ or\ equal\ to\ MySQL\ Database$$

$$H_1: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ READ\ operation\ is\ better\ than\ MySQL\ Database$$

For this hypothesis test, the p-values is 3.895×10^{-10} and it is less than alpha. Therefore, we conclude that Firebase Realtime Database response time performance when doing READ operation is better than MySQL Database. This operation allow us to evaluate the application responsiveness when parents want to check their children nutritional status, child growth chart, nutritional status measurement history, parents data, nutritional content of each food ingredients, daily toddlers food list, and food recommendation on our Toddlers Daily Nutritional Application.

7.3. UPDATE Operation

The tests for *UPDATE* operations that update 3000 records previously inserted into the database. This operation represents the application responsiveness in conditions when parents want to update their child data, parents data, and daily toddlers food list in our Toddlers Daily Nutritional Status Application. The results for these test are presented in the Figure 4 that illustrates the performance results between Firebase Realtime Database and MySQL Database of UPDATE operation. Y Axis shows response time in millisecond and X Axis shows amount of data, from the graphic illustrated that MySQL produce slower response time than Firebase Realtime Database.

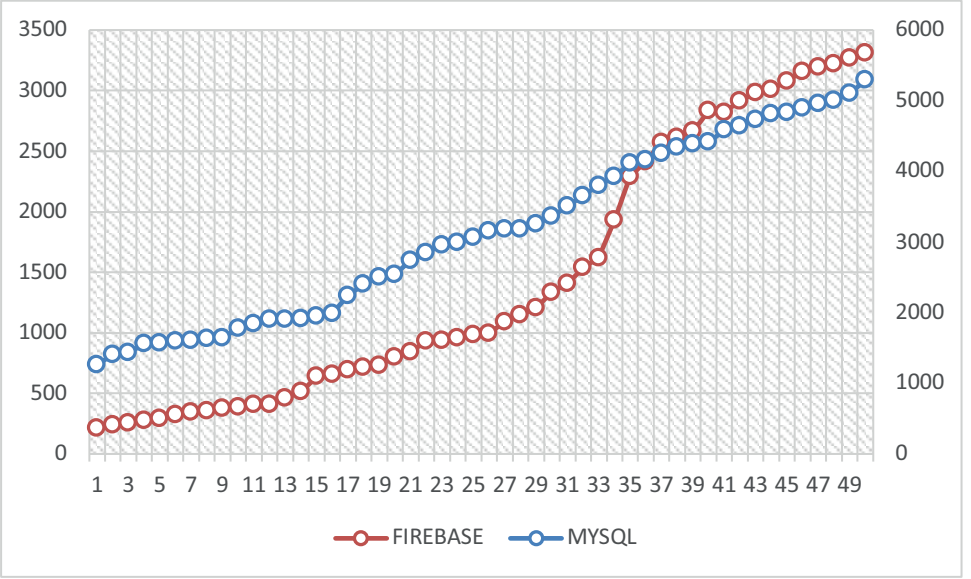

Fig. 4. Mean performance time in milliseconds for the UPDATE operation

Here is the hypothesis for UPDATE operation.

$$H_0: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ UPDATE\ operation\ is\ worse\ than\ or\ equal\ to\ MySQL\ Database$$

$$H_1: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ UPDATE\ operation\ is\ better\ than\ MySQL\ Database$$

The p-value for this test is 3.89×10^{-10}. Because the p-value < alpha, we reject the null hypothesis. It means that the performance of Firebase Realtime Database is better than MySQL Database in term of UPDATE operation.

7.4. DELETE Operation

The test for DELETE operation is to remove the previously inserted 3000 records into the database. This operation represents the application responsiveness in conditions when parents want to remove their child data, parents data, and daily toddlers food list in our Toddlers Daily Nutritional Status Application. The results for these test are presented in the Figure 5 that illustrates the performance results between Firebase Realtime Database and MySQL Database of DELETE operation. Y Axis shows response time in millisecond and X Axis shows amount of data, from the graphic illustrated that MySQL produce slower response time than Firebase Realtime Database.

Here is the hypothesis for DELETE operation.

$$H_0: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ DELETE\ operation\ is\ worse\ than\ or\ equal\ to\ MySQL\ Database$$

$$H_1: Firebase\ Realtime\ Database\ response\ time\ performance\ doing\ DELETE\ operation\ is\ better\ than\ MySQL\ Database$$

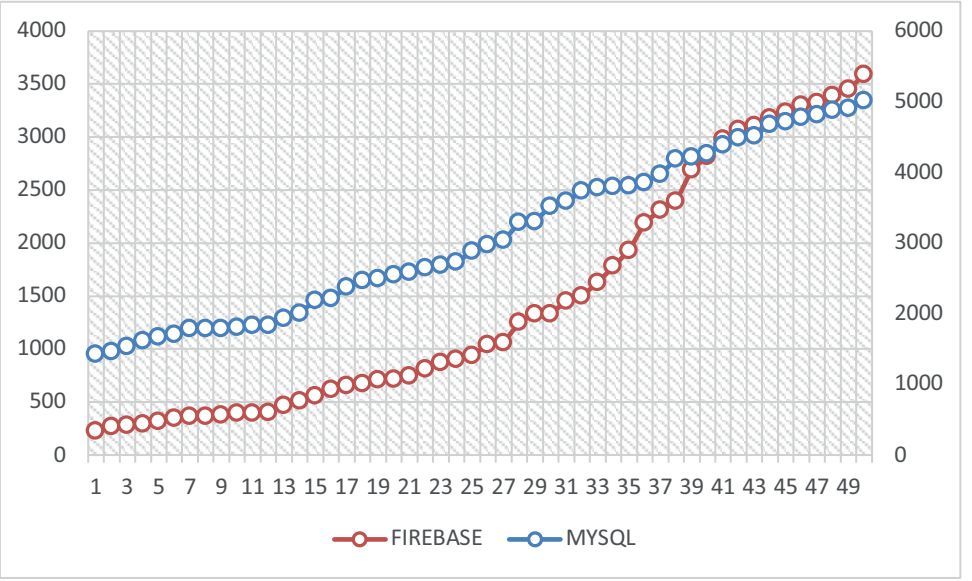

Fig. 5. Mean performance time in milliseconds for the DELETE operation

The p-value for this test is 3.892×10^{-10}. Because the p-value < alpha, we reject the null hypothesis. It means that the performance of Firebase Realtime Database is better than MySQL Database in term of DELETE operation.

From all CRUD operations, Firebase Realtime Database shows better performance rather than MySQL Database. Figure 2 and Figure 3 show that the performance of MySQL is slower than Firebase. However, in Figure 4 and Figure 5, we can see that Firebase produce slower response time than MySQL in a few observations. This is why we use the Shapiro-Wilk test. The drawback of the Firebase as mentioned in section Introduction will not affected its performance. It is because the data that inputted in the application more simple.

8. Conclusion

Based on the Wilcoxon Signed-Rank test of the comparison between Firebase Realtime Database and MySQL Database, it can be concluded that Firebase Realtime Database is more suitable as our Daily Nutritional Needs Mobile Application *Database Management System*. The comparison is used CRUD operations. From each CRUD operations results obtained Firebase Realtime Database perform better response time compared with MySQL Database. Aside from the terms of performance, by comparing its Key Features Firebase Realtime Database more suitable for our Daily Nutritional Needs Mobile Application since. It is because the Firebase provide Realtime Data and Simplifies Backend Process that help our developer build the Mobile Application without worrying about the server-side code.

References

1. Coronel C, Morris S. Database Systems Design, Implementation, and Management. 12th ed. Boston: Cengage Learning; 2017.
2. Boucadair M, Jacquenet C. In Handbook of Research on Redesigning the Future of Internet Architectures.; 2015.
3. Van Der Lans RF. SQL for MySQL Developers A comprehensive Tutorial and Reference Boston: Pearson Education; 2007.
4. Branson T. Dataconomy. [Online].; 2017 [cited 2019 01 31. Available from: https://dataconomy.com/2017/04/5-reasons-challenges-mysql/.
5. Moroney L. The Definitive Guide to Firebase Build Android Apps on Google's Mobile Platform Washington:

Apress; 2017.

6. Losari A. Medium. [Online].; 2018 [cited 2019 02 01. Available from: https://medium.com/@alfianlosari/firebase-realtime-database-many-to-many-relationship-schema-4155d9647f0f.
7. Truică C, R˘adulescu F, Boicea A, Bucur I. Performance evaluation for CRUD operations in asynchronously replicated document oriented database. International Conference on Control Systems and Science. 2015;: p. 1-3.
8. Tang E, Fan Y. Performance comparison between Five NoSQL Databases. International Conference on Cloud Computing and Big Data. 2016.
9. Pereira D, de Morais W, de Freitas E. NoSQL real-time database performance comparison. International Journal of Parallel, Emergent and Distributed Systems. 2017.
10. Ohyver M, Moniaga JV, Yunidwi KR, Setiawan MI. Logistic Regression and Growth Charts to Determine Children Nutritional and Stunting Status: A Review. In Procedia Computer Science; 2017; Denpasar. p. 232-241.
11. Aczel AD, Sounderpandian J. Business Statistics. 7th ed.: McGraw-Hill/Irwin; 2008.
12. Li Y, Manoharan S. A performance comparison of SQL and NoSQL databases. 19th ed.; 2013.
13. Stonehem B. Google Android Firebase: Learning the Basics; 2016.